

Blender

3D Content Creation Noob to Pro

[Wikibooks.org](https://www.wikibooks.org)

July 1, 2012

On the 28th of April 2012 the contents of the English as well as German Wikibooks and Wikipedia projects were licensed under Creative Commons Attribution-ShareAlike 3.0 Unported license. An URI to this license is given in the list of figures on page 1775. If this document is a derived work from the contents of one of these projects and the content was still licensed by the project under this license at the time of derivation this document has to be licensed under the same, a similar or a compatible license, as stated in section 4b of the license. The list of contributors is included in chapter Contributors on page 1737. The licenses GPL, LGPL and GFDL are included in chapter Licenses on page 1851, since this book and/or parts of it may or may not be licensed under one or more of these licenses, and thus require inclusion of these licenses. The licenses of the figures are given in the list of figures on page 1775. This PDF was generated by the \LaTeX typesetting software. The \LaTeX source code is included as an attachment (`source.7z.txt`) in this PDF file. To extract the source from the PDF file, we recommend the use of <http://www.pdflabs.com/tools/pdftk-the-pdf-toolkit/utility> or clicking the paper clip attachment symbol on the lower left of your PDF Viewer, selecting Save Attachment. After extracting it from the PDF file you have to rename it to `source.7z`. To uncompress the resulting archive we recommend the use of <http://www.7-zip.org/>. The \LaTeX source itself was generated by a program written by Dirk Hünninger, which is freely available under an open source license from http://de.wikibooks.org/wiki/Benutzer:Dirk_Huenniger/wb2pdf. This distribution also contains a configured version of the `pdflatex` compiler with all necessary packages and fonts needed to compile the \LaTeX source included in this PDF file.

Contents

I.	Unit 1 - Knowing Before Making	3
1.	Knowing Before Making	5
1.1.	Additional Resources	7
2.	What Blender Can Do	9
2.1.	Additional Resources	10
3.	3D Geometry	13
3.1.	Coordinates And Coordinate Systems	13
3.2.	Axes Of Rotation	15
3.3.	Additional Resources	17
4.	Coordinate Transformations	19
4.1.	Coordinate Transformations	19
4.2.	Linear Transformations	21
4.3.	Multiple Transformations	22
4.4.	Inverse Transformations	24
4.5.	Projections	25
5.	Orthographic Views	27
5.1.	Example	27
5.2.	Additional Resources	29
6.	Perspective Views	31
6.1.	One-point Perspective	32
6.2.	Two-point Perspective	33
6.3.	Three-point Perspective	34
6.4.	Additional Resources	34
7.	Coordinate Spaces in Blender	35
7.1.	Global and local coordinates	36
7.2.	View coordinates	38
7.3.	Normal coordinates	39
8.	User Interface Overview	41
9.	Keystroke, Button, and Menu Notation	43
9.1.	Hotkeys	43
9.2.	Mouse Notation	45
9.3.	Navigating Menus	45
9.4.	Additional Resources	47
10.	Non-standard equipment	49
11.	Operating System specific notes	51
11.1.	GNU+Linux	51
11.2.	Mac OS X	52
11.3.	Microsoft Windows	52
12.	Blender Interface	55

13.	Blender Windowing System	57
13.1.	An Interface Divided	57
13.2.	Window Headers	58
13.3.	Window Types	59
13.4.	The Active Window	64
13.5.	Resizing Windows	64
13.6.	Splitting a Window	65
13.7.	Joining Windows	66
13.8.	Leaving Blender	67
13.9.	Additional Resources	68
14.	User Preferences Windows	69
14.1.	Saving User Preferences	69
14.2.	Accessing the User Preferences	69
14.3.	Configuring Your Preferences	70
14.4.	Additional Resources	73
15.	Buttons Windows	75
16.	The Contexts	77
16.1.	Render Context File:Blender255RenderContextButton.png	77
16.2.	Scene Context File:Blender255SceneContextButton.png	77
16.3.	World Context File:Blender255WorldContextButton.png	77
16.4.	Object Context File:Blender255ObjectContextButton.png	77
16.5.	Object Constraints Context File:Blender255ObjectConstraintsContextButton.png	77
16.6.	Object Modifiers Context File:Blender255ObjectModifiersContextButton.png	77
16.7.	Object Data Context File:Blender255ObjectDataContextButtons.png	78
16.8.	Material Context File:Blender255MaterialContextButton.png	78
16.9.	Texture Context File:Blender255TextureContextButton.png	78
16.10.	Particles Context File:Blender255ParticlesContextButton.png	78
16.11.	Physics Context File:Blender255PhysicsContextButton.png	78
17.	Where Did The Old Stuff Go?	79
18.	3D View Windows	81
18.1.	The Viewport and its Contents	81
18.2.	Modes	86
18.3.	Viewport Options	88
18.4.	Undo	88
18.5.	Changing Your Viewpoint, Part One	89
18.6.	Positioning the 3D Cursor	91
18.7.	Changing Your Viewpoint, Part Two	94
18.8.	Visibility Layers	95
18.9.	Adding and Deleting Objects	97
18.10.	Count Your Polys	97
19.	Blender Interface (version 2.5)	99
20.	Blender Windowing System (version 2.5)	101
20.1.	The Blender Workspace	101
20.2.	The Active Window	104
20.3.	Resizing A Window	104
20.4.	Shelves	104
20.5.	Too Much To Fit	105
20.6.	Panels	106

20.7.	Splitting And Joining Windows	106
20.8.	The Default Workspace	108
20.9.	Workspace Presets	108
21.	User Preferences Window (version 2.5)	111
21.1.	Saving User Preferences	111
21.2.	Accessing User Preferences	111
21.3.	Configuring Your Preferences	111
22.	Properties Window (version 2.5)	115
23.	3D View (version 2.5)	117
24.	Blender Memory Management	119
24.1.	Datablocks And Users	119
II.	Unit 2 - Basic Modeling and Shading	123
25.	What is a Mesh?	125
25.1.	Additional Resources	132
26.	Quickie Model	133
26.1.	Bring up the Default Cube	134
26.2.	Setting up the Viewport	135
26.3.	Adjusting the Height	137
26.4.	Extruding	139
26.5.	Merging	140
26.6.	Saving your Work	142
26.7.	Additional Resources	142
27.	Quickie Render	143
27.1.	Rendering the Quickie Model	143
27.2.	Aiming the Camera	144
27.3.	Lighting	144
27.4.	Saving the Render	145
27.5.	Image File Formats	145
27.6.	Additional Resources	145
28.	Understanding the Camera	147
29.	Improving Your House	151
29.1.	Adding a Ground Plane	152
29.2.	Scaling with a Pivot	154
29.3.	Edge Selection	155
29.4.	Extruding Edges	156
29.5.	Face Selection	158
29.6.	Extruding Faces	160
29.7.	Deleting Edges	162
29.8.	Subdividing Faces	162
29.9.	Final Steps	165
30.	Modeling a Simple Person	167
30.1.	Start a New Scene	168
30.2.	Selection Methods	168
30.3.	Extruding Limbs	172
30.4.	Adding the Head	178
30.5.	Save Your Work	180
31.	Detailing Your Simple Person 1	183
31.1.	Subsurfaces	184

31.2.	Smooth Shading	187
32.	Detailing Your Simple Person 2	191
32.1.	Widening the Torso	191
32.2.	Bending the Arms	193
32.3.	Making Feet	195
32.4.	Reshaping the Head	195
33.	Creating a Simple Hat	199
33.1.	Creating a Generatrix	199
33.2.	Spinning the Hat	201
33.3.	Smoothing Your Hat	202
33.4.	Additional Resources	202
34.	Putting Hat on Person	203
34.1.	Adjusting an Object's Median Point	204
34.2.	Positioning the Hat	205
34.3.	Parenting the Hat to the Person	207
34.4.	Renaming Objects	208
34.5.	Outliner Windows	209
34.6.	Good on ya' mate!	210
34.7.	Additional Resources	210
35.	Materials and Textures	211
35.1.	Material versus Texture	211
35.2.	Other Material Settings	211
35.3.	Types of Textures	212
35.4.	Additional Resources	212
36.	Quickie Material	213
36.1.	Your First Material	214
36.2.	Specifying Colors	217
36.3.	Additional Resources	220
37.	Multiple Materials Per Object	221
37.1.	Set the scene	222
37.2.	Colorize Time	222
38.	Image Textures	227
38.1.	Easy Image Texture Creation With Gimp	227
38.2.	Other Image Texture Editors	229
38.3.	Mapping an Image Texture to an Object	229
39.	Procedural Textures	231
39.1.	A Simple Wood Texture	231
40.	Quickie Texture	235
40.1.	Adding a texture	235
40.2.	Adding a Stucci texture	240
41.	Halo Materials	241
41.1.	Introduction	241
41.2.	Applying a Halo Material	242
41.3.	Adding/Modifying extra effects	244
42.	Aligned to view issue	251
43.	Bones	253
43.1.	Laying down bones	254
43.2.	Add a bone	255

43.3.	Extrude a second Bone	257
43.4.	Name the bones	258
43.5.	Parent the bones	258
43.6.	Moving the Bones	261
44.	Mountains Out Of Molehills	263
44.1.	Creating a simple plane	263
44.2.	First mountain	264
44.3.	Peaks vs. hills	267
44.4.	Shaping the world	269
44.5.	Smoothing things out	271
44.6.	Naturalness	272
44.7.	Adding your guy with the hat	273
45.	Modeling a volcano	275
45.1.	Adding a Plane	275
45.2.	Making the Mountain	275
45.3.	Forming the Crater	279
45.4.	Subsurf	281
45.5.	Adding Magma	284
45.6.	Varying the Terrain	285
46.	Penguins from spheres	289
46.1.	Setup	289
46.2.	Creating the body	290
46.3.	Shaping the head	291
46.4.	Extruding the wings	296
46.5.	Smoothing the wings	299
46.6.	Cutting the underside	302
46.7.	Adding the feet	304
46.8.	Extruding a tail	307
46.9.	Subsurfing	309
46.10.	Extra	310
47.	Die Easy 2	313
47.1.	Start with a beveled cube	313
47.2.	Divide the faces	314
47.3.	Colouring the Cube	323
47.4.	Adding the spots	324
47.5.	Subsurfing and smoothing	330
47.6.	Colouring the spots	334
47.7.	Afterword	338
48.	Die Another Way	339
48.1.	Video Tutorial	339
48.2.	Introduction	339
48.3.	Subdivide First	340
48.4.	Manual Sizing of Pips	340
48.5.	Creating Pips	345
48.6.	Color	357
49.	Model a Goblet	369
49.1.	Three ways to build the mesh	369
49.2.	How many circles?	370

50.	Model a Silver Goblet	373
50.1.	Basic Shape	373
50.2.	Smoothing and Defining	377
50.3.	Transparency	380
51.	Model a Silver Goblet cylinder	383
51.1.	Preliminaries	383
51.2.	Description of the modeling steps	383
51.3.	Creating the Goblet	385
51.4.	Subsurfing and smoothing the goblet	386
51.5.	Flattening the base of the goblet	387
52.	Spin a goblet	389
52.1.	Modeling	390
52.2.	Rendering	394
52.3.	Detailed keystrokes	395
53.	Light a Silver Goblet	397
53.1.	Techniques	397
53.2.	Objects in the Scene	397
53.3.	Adding the Atmosphere	400
54.	Simple Vehicle	407
55.	Simple Vehicle: Wheel	409
55.1.	Techniques	409
55.2.	Model the tire	409
55.3.	Model the hubcap	416
55.4.	Extra	421
56.	Simple Vehicle: Seat	423
56.1.	Techniques	423
56.2.	Extrude the Seat	423
56.3.	Add cushion seams	424
56.4.	Position the cushion seams	425
56.5.	Add Depth to the seams	426
56.6.	Subsurf the seat	428
56.7.	Resize the seat	429
56.8.	Final touches	430
57.	Simple Vehicle: Rocket Launcher	433
57.1.	Techniques	433
57.2.	Overview	433
57.3.	Create the Launcher	433
57.4.	Create the rocket	435
57.5.	Create the mount	437
57.6.	Subsurf	439
57.7.	Final touches	441
58.	Simple Vehicle: Body	443
58.1.	Techniques	443
58.2.	Planning	443
58.3.	Building the Jeep	443
58.4.	A Touch of Detail	463
58.5.	Subsurf	473
58.6.	Optional Activities	473

59.	Simple Vehicle: Some Assembly Required	475
59.1.	Techniques	475
59.2.	Overview	475
59.3.	Appending the File	475
59.4.	Rinse and Repeat	477
59.5.	Parenting	478
59.6.	Final things	479
59.7.	Extra	480
59.8.	Additional Tutorials	481
60.	Modeling a 3D Parachute in Blender	483
60.1.	Remove the default cube	483
60.2.	Add a cylinder	484
60.3.	Remove the bottom row of vertices	485
60.4.	Extrude and scale to shape	486
60.5.	Make the top more rounded	487
60.6.	Extrude the parachute straps	488
60.7.	Merge it together	489
61.	Model a Low Poly Head	491
61.1.	Overview	491
61.2.	Add a plane	491
61.3.	Make a pointed chin	492
61.4.	Extrude the face	494
61.5.	Make the facial features	496
61.6.	Smooth the mouth region	503
61.7.	Make the head	504
61.8.	Make the back of the head	506
61.9.	Finishing it up	510
62.	Building a House	511
62.1.	Setting the Scene	511
62.2.	Make the house walls	514
62.3.	Make the roof	515
62.4.	Make a chimney	523
62.5.	Adding a Window and a Door	525
62.6.	Building the Fence	532
62.7.	Put the Horizontal Bars Around The House (Method 2 (above) cont'd)	538
62.8.	The Ground plane and a Path	539
62.9.	Final render	540
63.	Pipe joints	543
63.1.	T joint	543
63.2.	6 cylinder joint	546
63.3.	3 cylinder joint	548
63.4.	T joint with smaller diameter	550
64.	Aligning Vertices with a Guide Image	553
64.1.	Background: orthographic projection	553
64.2.	Making a Simple Pyramid	555
64.3.	Using the guide images	556
65.	Modeling a Fox from Guide Images	567
65.1.	Method 1	567

65.2.	Method 2 (using mirroring)	581
65.3.	Using mirroring with method 1	583
65.4.	Detailed steps to align images using GIMP	584
66.	2D Image (logo) to a 3D Model	585
66.1.	Overview	585
66.2.	Load the background image into blender	585
66.3.	Delete the cube	592
66.4.	Introducing the Bezier Curve	592
66.5.	Modeling the lighting bolt	598
66.6.	Modeling the red circle	607
66.7.	Extra Practice	626
67.	Curve and Path Modeling	627
68.	Intro to Bezier Curves	629
68.1.	Bezier curve basics	629
68.2.	Extruding from 2d to 3d	632
68.3.	Create a filled, extruded shape	638
69.	Curves In 3d	639
69.1.	Bezier Curves in 3d	639
69.2.	Getting started in 3D	640
69.3.	Extruding revisited	640
69.4.	Curve Tilting and Normal-scaling	641
69.5.	Convert Curve to Mesh	641
70.	Deforming Meshes using the Curve Modifier	643
70.1.	Setting up your Mesh	643
70.2.	Extruding the Cone	644
70.3.	Making a Curve	645
70.4.	Applying the Curve to the Mesh	647
71.	Making Your Creation Smoother	649
71.1.	Getting started	649
71.2.	Sculpting With "Lumps of Clay"	649
71.3.	Meta-mess!	651
71.4.	Beginning to sculpt	653
72.	Using Textures	655
72.1.	Colormap	655
72.2.	Diffusemap	657
72.3.	Luminositymap	659
72.4.	Specularitymap	659
72.5.	Reflectionmap	660
72.6.	Transparencymap	660
72.7.	Refractionmap	661
72.8.	Translucencymap	661
72.9.	Bumpmap/Normalmap	662
72.10.	Displacement Map	664
72.11.	Dirtmap	664
72.12.	Use UV Coordinates for your Maps	667
73.	Using a texture to make a material partially transparent	669
73.1.	Using a greyscale texture - white for opaque	670
73.2.	Using a partially transparent texture to make the material transparent	672

74.	Creating Basic Seawater	675
74.1.	Create 3 linked planes	676
74.2.	Create material	676
74.3.	Extra Practice	678
75.	Mountains Out Of Molehills 2	679
76.	Basic Carpet Texture	683
76.1.	Goal	683
76.2.	The Basic Material and Shader Settings	684
76.3.	Cloud Texture 1	685
76.4.	Cloud Texture 2	686
76.5.	The Final Cloud Texture	687
76.6.	The Last Texture	689
77.	The Rusty Ball	691
78.	Procedural Eyeball	693
78.1.	Creating the Mesh	694
78.2.	Adding the Textures	695
78.3.	Pupil Dialation	700
79.	Creating Pixar-looking eyes in Blender	703
79.1.	Parts of the Eye	704
79.2.	Materials	705
79.3.	Lighting	709
79.4.	Changing the Eye Color	709
79.5.	Pixar Eye Video Tutorial	713
79.6.	Useful Links	713
80.	UV Map Basics	715
80.1.	The Basics of UV Mapping	715
80.2.	Questions	724
80.3.	Alternative method	726
80.4.	Source	728
80.5.	Texface versus Map input - pasting a photo to a sphere	729
80.6.	Video Tutorial	731
81.	Realistic Eyes In Blender	733
81.1.	Overview	733
81.2.	Reference Material	734
81.3.	Building The Eye	735
81.4.	UV Mapping	740
81.5.	Preparing A Template For Texturing	744
81.6.	First Bump Map - The Sclera	746
III.	Unit 3 - Broadening Horizons	751
82.	Beginning Lighting	753
83.	Understanding Real Lights	755
83.1.	The properties of real lights	755
84.	Adding Lamps	757
84.1.	Explaining the Different Lights:	757
84.2.	Creating a basic scene with basic lighting	758
85.	Faked Gi with Blender internal	769
85.1.	Preamble	769
85.2.	Blender Faked GI Tutorial	770

86.	Render Settings	775
86.1.	The Scene Tab	775
86.2.	Output / Render Layers	776
86.3.	Render	779
86.4.	Anim / Bake	780
87.	Output Format Options	783
87.1.	Output Format Basics	783
87.2.	Output formats	785
87.3.	Cropping	786
87.4.	Palette	786
87.5.	Pre-sets Just For You	787
88.	OSA	789
89.	Looking All Around - Panorama Settings	791
89.1.	Panoramic Renderings	791
90.	Other Important Render Options	797
91.	YafRay Render Options	799
91.1.	Getting Started with YafRay	800
91.2.	YafRay tab	801
92.	Stamping	803
92.1.	Tictoon Tutorial: Stamping	803
93.	2D, Toon, & CAD Rendering	811
93.1.	2D, Toon, & CAD Rendering	811
93.2.	Freestyle & Blender Integration	812
94.	Basic Animation	815
95.	ScreenLayout	817
96.	Lattice	821
96.1.	What is a Lattice?	821
96.2.	How to add and use a Lattice	821
96.3.	Getting more involved with Lattice	823
96.4.	Examples	827
96.5.	Making it stick	828
96.6.	Animating a Lattice Modifier	828
96.7.	Let the fun begin!	830
97.	Bounce	833
97.1.	How to make a ball bounce convincingly!	833
97.2.	How hard can it be to move a ball?	833
97.3.	Why use a lattice?	833
97.4.	Let's begin then	833
97.5.	Saving The Animation	842
97.6.	So what next?	842
97.7.	Links	842
98.	Creating Basic Water animation	843
98.1.	Water and Other Fluids	843
98.2.	The Domain	844
98.3.	The Fluid	844
98.4.	The Obstacle	844
98.5.	Baking Fluids	845
98.6.	Other Fluid Objects	846

98.7.	Keep This In Mind	846
98.8.	Notes	846
98.9.	Site Under Construction	846
99.	Flying Through A Canyon	847
99.1.	Creating the Canyon	847
99.2.	Coloring the Canyon	852
99.3.	Additional projects	853
100.	Using the sequencer to compile frames into an animation	855
100.1.	Using the sequencer to compile frames into an animation	855
100.2.	Preamble	855
100.3.	Tutorial	855
100.4.	Use the Sequencer to compile the images into a movie file	857
101.	Particle Systems	861
101.1.	The very first particle system	861
101.2.	Links	865
102.	Making Fire	867
102.1.	The particle system	868
102.2.	Material	871
102.3.	Rendering	873
103.	Furry	875
103.1.	The emitter	876
103.2.	Material	879
103.3.	Changing Hair length with a texture	882
103.4.	Comb it!	885
103.5.	Links	886
104.	Fireworks	887
104.1.	The Emitter	887
104.2.	Reactor 1	888
104.3.	Smoke Trail	890
104.4.	Render	891
105.	Particles forming Shapes	893
105.1.	Keyed Physics	893
105.2.	Harmonic Force Fields	898
106.	Billboard Animation	901
106.1.	Splitting a texture	902
106.2.	Animating Alpha in relative particle time	904
106.3.	Animating billboard color	906
106.4.	Changing the starting color of a billboard	907
107.	Soft Body Animation	909
107.1.	Explanation of Settings	913
108.	Simple Cloth Animation	915
108.1.	Making the Skirt Mesh	915
108.2.	Creating the Vertex Group	915
108.3.	Animating the Skirt	915
108.4.	Keeping the Folds	916
108.5.	Extra Practice	917
109.	Soft Body with wind	919
109.1.	Setting up scene	919

109.2.	Designating each object's job	921
109.3.	The movement	924
109.4.	Finishing touches	927
109.5.	Final result	930
110.	Your First Test	933
110.1.	Adding the Hill	933
110.2.	Creating the Ball	933
110.3.	Testing your game	934
110.4.	Video capturing your game	935
110.5.	Conclusion	936
110.6.	Extra Tutorials	936
111.	Platformer: Creation and Controls	937
111.1.	Set Up	937
111.2.	Controls	938
112.	An aMAZEing game engine tutorial	941
112.1.	Maze Surface	941
112.2.	Character	944
112.3.	Motion	946
112.4.	Testing	947
112.5.	Falling	948
112.6.	Camera	950
112.7.	Beautification	950
112.8.	Levels	954
112.9.	Dynamic obstacles	955
112.10.	Conclusion	957
113.	Platformer: Physics Fixes	959
113.1.	Creating Your Hit Test Object	959
113.2.	Excessive Bouncing	960
113.3.	Final Notes	960
114.	Making exe	961
114.1.	Windows	961
114.2.	GNU/Linux or Mac OS X	963
114.3.	BlenderPlayer	963
114.4.	Legal Issues	963
114.5.	Proprietary Blend Files	964
115.	Build a skybox	965
115.1.	Gather your graphics	965
115.2.	Create a dome for the sky	967
115.3.	Create a dome for the ground	971
115.4.	Render the environment map	973
115.5.	Video Tutorial	976
116.	Basic mouse pointer	977
117.	Text in BGE	979
118.	Python Platformer: Creation	981
118.1.	Code Sections	981
118.2.	Adding the Player Object	982

IV. Unit 4 - Taking Off With Advanced Tutorials	985
119. Python Scripting	987
119.1. Chapter contents	987
120. Introduction	989
120.1. Introduction	989
120.2. First Steps In Blender Scripting	989
120.3. Multiple Running Scripts	990
120.4. The Blender Versus bpy Modules	990
120.5. See Also	991
121. Introduction (version 2.5)	993
121.1. First Steps In Blender Scripting	993
121.2. The bpy Module	994
121.3. The mathutils Module	994
121.4. See Also	995
122. Addon Anatomy (version 2.5)	997
122.1. Text Blocks	997
122.2. Your First Operator	997
122.3. Invoking Your Operator	1000
122.4. If You Hit An Error	1000
123. Addon User Interface (version 2.5)	1003
123.1. Your First Panel	1003
123.2. Adding Undo Support	1003
123.3. Put It All Together	1004
123.4. Space Types, Region Types, Contexts, Oh My!	1005
124. Custom Property (version 2.5)	1007
124.1. Defining A Property	1007
124.2. Using The Property	1008
124.3. Put It All Together	1008
125. Separately Installable Addon (version 2.5)	1011
125.1. The Addon Info Dictionary	1011
125.2. Making The Addon Installable	1012
125.3. Installing The Addon	1012
125.4. Using The Installed Addon	1012
126. Object, Action, Settings (version 2.5)	1015
126.1. Adding To The Add Menu	1015
126.2. Fixing Up The UI	1016
126.3. invoke Versus execute	1017
126.4. Put It All Together	1017
126.5. Undocumented Blender	1018
127. Procedural object creation	1033
127.1. Text Blocks	1033
127.2. Your First Script	1033
127.3. Put It All Together	1034
127.4. If You Hit An Error	1035
128. External scripts	1037
128.1. The Script Header	1037
128.2. Put It All Together	1037
128.3. Making The Script Installable	1038

128.4.	Installing The Script	1038
128.5.	Using The Installed Script	1039
129.	Creating a GUI for your script	1041
129.1.	A Modal GUI	1041
129.2.	A Modeless GUI	1042
130.	Export scripts	1045
130.1.	Finding out about things in a scene	1046
130.2.	Creating a script	1049
130.3.	Exporting a Mesh	1052
131.	Import scripts	1055
131.1.	Importing a Mesh	1055
132.	Scripts for modifying meshes	1057
133.	Advanced Modeling Introduction	1059
133.1.	Advanced Modeling	1059
134.	Advanced Modeling	1061
134.1.	Making Better Models	1061
135.	HDRi	1063
135.1.	Introduction	1063
135.2.	Definitions	1063
135.3.	Usage	1064
135.4.	Quick Tutorial (<i>for experienced blendies</i>)	1065
135.5.	Step-by-step Tutorial	1065
136.	Creating a Light Probe	1071
137.	Making Landscapes with heightmaps	1075
137.1.	Creating the heightmap image	1075
137.2.	Create grid and add the image as texture	1076
137.3.	Use texture as heightmap	1078
138.	How to Do Procedural Landscape Modeling	1083
138.1.	Mountains	1083
139.	Landscape Modeling I: Basic Terrain	1089
139.1.	Creating the Canvas	1089
139.2.	Molding the Mountains	1090
139.3.	Texturing the Terrain	1093
139.4.	Reader Contributions	1095
139.5.	Reader Contributions 2	1096
139.6.	Reader Contributions 3	1097
140.	Landscape Modeling II: Texture Stenciling	1099
140.1.	Creating the Stencil	1099
140.2.	Applying the Stencil	1102
140.3.	Adding Snow	1104
140.4.	Multiple Stencils	1104
141.	Landscape Modeling III: Exporting as a Heightmap	1109
141.1.	A Word About Heightmaps	1109
141.2.	Creating the Material	1111
141.3.	Setting up the Camera	1113
141.4.	Avoid Stair-Stepping	1115
142.	Realistic Water using Fluid Sim and Yafray	1117
142.1.	Using the Fluid Simulator and Yafray to create realistic looking water	1117

142.2.	Getting Started	1117
142.3.	Extra Practice	1123
143.	Bump Mapping	1125
143.1.	Creating Bump Maps	1125
143.2.	Applying the animated bump map	1129
144.	Normal Mapping	1131
145.	Texture Normal Mapping	1133
145.1.	The Texture-Normal	1133
146.	Color Map Normal Mapping	1137
146.1.	The Color-Map-Normal	1137
146.2.	Creating a Normal Map	1137
147.	Nodes	1143
148.	Texture Nodes	1145
148.1.	A Simple “Rainbow” Texture	1145
148.2.	Scalars Versus Vectors	1147
149.	Material Nodes	1149
149.1.	A Simple Graduated Material	1149
150.	Compositing	1153
150.1.	Introduction to Composite Nodes	1153
151.	Introduction	1155
152.	Armature Object	1157
153.	Object View	1159
153.1.	The Armature Object	1159
153.2.	The Edit Panel When in Object Mode	1161
153.3.	Extra Practice	1169
154.	Edit View	1171
154.1.	Now the basics about bones	1172
154.2.	The edit panel	1173
154.3.	Naming convention	1179
154.4.	Mirror Editing	1180
155.	Pose View	1183
155.1.	So What Can You Do?	1183
156.	Mesh Object	1187
157.	Armature and Mesh	1189
157.1.	The Armature Modifier	1189
157.2.	The Old Way	1190
157.3.	Tip: Bake envelope to vertex groups	1191
158.	Envelope	1193
158.1.	What is Envelope	1193
158.2.	Edit Envelope	1194
158.3.	Envelope Options	1195
159.	Vertex Groups and Weight Paint	1199
159.1.	What Are Vertex Groups?	1199
159.2.	Weight Paint	1200
159.3.	Vertex Groups and Armatures	1202
159.4.	Using Weight Painting with Armatures	1203
160.	Shape Key	1207
160.1.	Shape Key?	1207

160.2.	The GUI	1210
160.3.	Shape Keys step-by-step	1210
161.	Lip-Sync	1213
161.1.	Lip-Sync with Shape Keys	1213
161.2.	The hard work first	1213
161.3.	Getting down and dirty	1219
161.4.	Putting it all together	1222
161.5.	Using Blender to render Audio AND Video TOGETHER.	1222
161.6.	Audio Files	1223
161.7.	P.S.	1223
162.	Constraints	1225
162.1.	The Constraint	1225
162.2.	The Constraint Panel	1225
162.3.	The Constraint Index	1227
163.	Copy Location	1229
163.1.	Copy Location	1229
163.2.	The Constraint Panel	1230
163.3.	Where To Use It	1230
164.	Copy Rotation	1233
164.1.	Copy Rotation	1233
164.2.	The Constraint Panel	1234
164.3.	Where To Use It	1234
165.	Track-To	1235
165.1.	Track-To	1235
165.2.	The Constraint Panel	1236
165.3.	Where To Use It	1237
166.	Floor	1239
167.	Stretch-To	1241
167.1.	Stretch-To	1241
168.	IK Solver	1243
168.1.	The IK solver	1243
168.2.	The Constraint Panel	1244
168.3.	Where To Use It	1248
168.4.	Degree Of Freedom	1249
169.	NLA Window	1251
170.	(about this placeholder)	1253
171.	The NLA Window	1255
171.1.	Forum Notes	1255
171.2.	Walkthrough	1255
172.	Intro to the NLA Editor	1259
172.1.	NLA (Non-Linear Animation)	1259
172.2.	Setting up the Scene	1259
172.3.	Adding Action Strips	1260
172.4.	Getting the Strips to Play	1263
172.5.	Conclusion	1265
173.	The Stride Feature	1267
174.	Relative Vertex Keys	1269

175.	Piston, Rod and Flywheel	1271
175.1.	Piston, Rod and Flywheel	1271
175.2.	Links	1278
176.	Cutting Through Steel	1281
176.1.	Creating the Alpha-Map	1282
176.2.	Welding sparks	1285
176.3.	Using the alpha map on an object	1286
176.4.	Material for the plane	1289
176.5.	Lighting the torch	1290
176.6.	More realistic sparks	1291
176.7.	Smoke	1292
176.8.	After Glow	1295
176.9.	Links	1297
177.	Advanced Game Engine	1299
178.	Game Creating Techniques Part One: The GUI	1301
178.1.	Making Blender Games Graphically	1301
179.	Creating Pop-Up Menus	1303
179.1.	Making An Easy Title Menu	1303
179.2.	Author's Starting Menu Example	1305
179.3.	A Main Menu	1306
180.	Creating Moving Menus	1311
180.1.	Making Moving Selectors	1311
180.2.	Make the Menu Move	1315
181.	The "5-Layer"Button	1317
181.1.	Scheduled Construction	1317
182.	Creating Object Outlines	1319
182.1.	Scheduled Construction	1319
183.	Game Creating Techniques Part Two: Python	1321
183.1.	Needs Expert	1321
183.2.	Additional Resources	1321
183.3.	External Links	1321
184.	Really Advanced: The Game Engine Source	1323
V.	Appendices	1325
185.	Glossary	1327
185.1.	A	1327
185.2.	B	1327
185.3.	C	1328
185.4.	D	1328
185.5.	E	1329
185.6.	F	1329
185.7.	G	1330
185.8.	H	1330
185.9.	I	1330
185.10.	J	1331
185.11.	K	1331
185.12.	L	1331
185.13.	M	1331
185.14.	N	1332

185.15.	O	1332
185.16.	P	1333
185.17.	Q	1333
185.18.	R	1333
185.19.	S	1334
185.20.	T	1335
185.21.	U	1335
185.22.	V	1336
185.23.	W	1336
185.24.	X	1336
185.25.	Y	1336
185.26.	Z	1336
186.	FAQ	1337
187.	Tutorial Links List	1339
187.1.	Official Blender Documentation	1339
187.2.	Interface	1339
187.3.	Mesh Modeling	1340
187.4.	Nurbs and Subsurface Modeling	1342
187.5.	Specific Object Modeling	1342
187.6.	Texture Mapping	1345
187.7.	2D Texture Painting Techniques	1347
187.8.	Lighting, Shadows and Rendering	1348
187.9.	Armatures and IK	1349
187.10.	Animation	1350
187.11.	Particles	1351
187.12.	Fluid Simulation	1352
187.13.	Compositing	1352
187.14.	Game Engine	1353
187.15.	Python and Plugins	1354
187.16.	Using other Programs with Blender	1355
187.17.	Video Tutorials	1357
187.18.	Blender WikiBooks	1357
187.19.	FAQ	1358
187.20.	Miscellaneous	1358
187.21.	Other Lists	1359
187.22.	About	1360
188.	Hotkeys	1361
189.	Output Formats	1363
190.	Image Portfolio	1365
190.1.	Summary	1365
190.2.	Categories	1365
190.3.	A sampling of images from the above categories	1365
191.	Blender Glossary	1411
192.	Every Material Known to Man	1415
192.1.	"Every Material Known To Man"Project	1415
193.	Sources of free 3D models	1417
193.1.	Blender-Specific	1417
193.2.	Non-Blender-Specific	1417

193.3.	Sites Using .RAR Format	1419
193.4.	Sites Requiring Registration	1419
194.	Blueprint Links List	1421
194.1.	Registration Required	1421
194.2.	Payment Required	1421
194.3.	Shouldn't Really Be Here?	1422
195.	Materials, Textures, Photos	1423
196.	Asking for Help	1425
197.	Know What You're Doing	1427
198.	Modeling Realistically	1429
198.1.	Assemble Things, Don't Chisel Them	1429
198.2.	Separate Objects for each Material	1430
198.3.	Removing the Fourth Wall	1431
198.4.	Proportions and Measurements	1431
199.	Modeling tips	1433
200.	Cheat the 3D	1435
200.1.	Ways to Improve Performance	1435
201.	Performance vs. Quality	1437
202.	Modeling a Gingerbread Man	1439
202.1.	Modeling	1439
202.2.	Camera Positioning and Rendering	1458
202.3.	Applying Textures	1459
203.	Modeling a simple space-ship	1465
203.1.	Modeling a simple spaceship using Blender 2.49b	1465
204.	Creating Weapons based on 2D Images	1467
204.1.	Modeling Technique 1	1467
204.2.	Modeling Technique 2	1468
205.	Match Moving	1471
205.1.	Icarus	1471
205.2.	Voodoo	1471
206.	Motion Tracking with Icarus	1473
207.	Create a Clayman	1477
207.1.	Create a Clayman	1477
207.2.	Making The Clayman	1477
207.3.	Adding Materials	1487
207.4.	Inserting Armatures	1488
207.5.	Applying The Armature	1491
208.	Organic Modeling	1505
209.	Understanding the Fluid Simulator	1509
209.1.	Understanding the Fluid Simulator	1509
209.2.	First of all	1509
209.3.	Setting up the scene	1509
209.4.	Setting up the simulation	1510
209.5.	Final notes	1511
209.6.	Extra Practice	1511
209.7.	Links	1512
210.	Creating a jewel in Blender	1513
210.1.	Modeling the jewel	1513

210.2.	Shading	1517
210.3.	The finishing touch	1518
210.4.	External links	1518
211.	Modeling a picture	1519
211.1.	Modeling a picture	1519
211.2.	Printing a Rendered Image	1520
212.	Modeling with the Spin Tool	1521
213.	Spin Tool Introduction	1523
213.1.	How the Spin Tool Works	1523
213.2.	Typical Work Flow	1524
213.3.	Post Modeling	1541
214.	Illustrative Example: Model a Wine Glass	1543
214.1.	Model half the outline	1543
214.2.	Spinning the outline	1544
214.3.	Cleaning up the Mesh	1547
215.	Creating Ogg-Theora movies using Blender	1549
215.1.	Converting saved frame picture files to Ogg Theora	1549
215.2.	Converting AVIs to Ogg Theora	1549
216.	Creating animated GIFs using Blender and Gimp	1551
217.	3D Tiling Backgrounds For The Web	1553
217.1.	Overview	1553
217.2.	Create The Object You Wish To Tile	1553
217.3.	Specify Your Tiling Area	1554
217.4.	Tile Away!	1555
217.5.	Camera Settings	1560
217.6.	Lighting & Materials	1561
217.7.	Rendering	1561
217.8.	'Shopping'	1561
218.	Cool Things	1567
218.1.	Attribution	1567
218.2.	File Browser Functions	1567
218.3.	Object/Vertex Manipulation	1568
218.4.	Working with Meshes	1574
218.5.	Animation	1575
218.6.	UV Mapping, Particles and Texturing	1575
218.7.	Rendering	1577
218.8.	Miscellaneous	1579
218.9.	???.	1581
218.10.	Sculpt Mode Hotkeys	1581
219.	Troubleshooting	1583
220.	Creating Blender Libraries	1585
220.1.	How to Make a Library	1585
220.2.	Library Usefulness Checklist	1585
220.3.	Publishing Your Library	1586
220.4.	Beyond Libraries	1587
220.5.	Thank You	1587
221.	Add some depth with stereo	1589
221.1.	The stereo camera	1589

221.2.	Stereo viewing with the rig	1592
221.3.	Stereo rendering	1592
222.	Fluffy Material	1597
222.1.	Spherical Blend Texture	1597
222.2.	Backlighting	1598
223.	Human Body	1603
224.	Using Blender Libraries	1607
224.1.	Indirect linking	1607
224.2.	Groups	1607
224.3.	See also	1608
225.	Beginning Modeling Final Project	1609
226.	Using Inkscape to make advanced Bezier curves	1611
226.1.	Introduction to Inkscape	1611
226.2.	Getting Started	1611
227.	Platonic Solids	1619
227.1.	External Link	1620
228.	Polygonal Modeling	1621
229.	Box Modeling	1623
229.1.	Before We Start	1623
229.2.	The Work Flow	1623
230.	Illustrative example: Model a Chair (Swan Chair)	1627
231.	Model a Chair-Preparations	1631
231.1.	Knowing our model	1631
231.2.	Loop study	1631
232.	Model a Chair-The Seat	1635
232.1.	Start with a Primitive	1635
232.2.	Rough it out	1635
232.3.	Adding More Details	1646
233.	Model a Chair-The Feet	1659
233.1.	Starting Primitive	1659
233.2.	The Upper Telescope	1660
233.3.	The Lower Telescope	1663
233.4.	The Stops	1674
233.5.	The Whole Chair	1684
234.	Illustrative Example: Modeling a Simple Human Character	1687
235.	Modeling a Human Character - Preparations	1689
236.	Modeling a Human Character - Modeling	1691
237.	Blocking the Figure	1693
238.	Detailing and Finishing	1701
239.	Polygon by Polygon modeling	1709
239.1.	Before we start	1709
239.2.	The Workflow	1710
240.	Animation Notes and FAQ	1717
240.1.	Blender 3D: Animation Notes and FAQ	1717
240.2.	IMPORTANT	1717
240.3.	Authors & Contributors	1717
240.4.	Introduction	1717
240.5.	Todo List	1718

240.6.	Animation FAQ	1718
240.7.	Armatures	1718
240.8.	Hotkeys	1718
240.9.	Approaches, Techniques, Tips and Tricks	1719
240.10.	Problems and Questions	1720
240.11.	General Armature Information	1723
240.12.	Building Armatures	1725
240.13.	Useful Rigs	1725
241.	Customization	1727
241.1.	Changing the Theme	1727
242.	Mist - Make Objects Opaque	1729
243.	Contributors	1737
	List of Figures	1775
244.	Licenses	1851
244.1.	GNU GENERAL PUBLIC LICENSE	1851
244.2.	GNU Free Documentation License	1852
244.3.	GNU Lesser General Public License	1852

Part I.

Unit 1 - Knowing Before Making

1. Knowing Before Making

Blender is a powerful and complex 3D modeling and rendering package. Before you can use it effectively to *make* things, you need to *know* a few things about how it works:

- the process of 3D modeling and rendering (what Blender does)
- some rudiments of 3D analytic geometry (axes and coordinates)
- orthographic and perspective views of a 3D object
- local coordinate systems and child objects
- the fundamentals of Blender's user interface (hotkeys, windows, and menus)
- how to view a 3D scene from different vantage points (in Blender)

This unit is entirely devoted to this sort of background knowledge. You won't create your first Blender model until the next unit.

Knowing this, you might be tempted to skip ahead. Depending on your background, this might or might not work. For instance, if you've used other 3D graphics , you might be able to skim (or skip) ahead as far as the user interface tutorial. But if there's any doubt, please proceed through the tutorials in sequence.

Blender is not the kind of software you can launch into and grope about until you find your way. It's not like exploring an unfamiliar city. It's more like flying a jet airplane. If you hop into the pilot's seat without knowing the fundamentals, you'll be lucky to ever get off the ground, and it'd take a miracle for you to reach your destination safely.

1.0.1. A Word about Jargon

Like any subject, 3D modeling has its own jargon: terminology specific to the subject and ordinary words that have special meanings in the context of computer graphics.

In this book, important new words are **highlighted** on first appearance and defined soon after. If you suspect you've missed (or forgotten) the meaning of a word, try looking it up in the Glossary¹.

1.0.2. Things You'll Need

In order to work through the tutorials, you'll need access to a computer that has Blender installed (download the latest stable release²).

Depending on what is installed on your system you may also need the appropriate Python and QuickTime installation. Each version of Blender works only with one specific version of Python,

¹ Chapter 191 on page 1411

² <http://www.blender.org/download/get-blender>

which is generally included in the download. For example, Blender 2.49 specifically required Python 2.6 and did not work with Python 2.7; Blender 2.5x generally uses Python 3.2.

Since Blender is open-source software, you can download the source code and build it for yourself, but it's easier to download a pre-built install package. Install packages are provided for each supported operating system:

- Microsoft Windows (32-bit and 64-bit)
- Linux (32-bit and 64-bit x86)
- Apple Mac OS X (PowerPC and Intel)
- FreeBSD (32-bit and 64 bit)
- Solaris
- Irix

Windows users get to choose between an executable installer ("setup wizard") and a ZIP archive. The installer is the easier route because it guides you through the install options. The default options are fine for beginners.

Blender uses the Python scripting language, so each install package comes bundled with a compatible version of Python. Some users can choose between different versions of Python (such as 2.5 and 2.6).

For quicker downloading, choose the download server nearest you:

- USA
- Germany
- NL (Netherlands)
- Asia

It's also a good idea to have pencil and paper handy for sketching and taking notes. There's a lot to absorb. Taking notes as you go will pay dividends later.

On a fully stable working Ubuntu³ 11.04 (Oneiric) or newer, or possibly other Linux distributions based on Debian, start by making sure the repository cache is updated if you haven't updated it within a day or so. This can be done in Update Manager or by entering the following command into a terminal:

- `sudo apt-get update`

`sudo apt-get` is an administrative command, so you'll probably need to enter your password. Once the repository cache is updated, you can download the stable release of Blender by entering the following command into a terminal:

- `sudo apt-get install blender`

Once that finishes, Blender should appear in the Graphics section of your desktop environment's application menu.

³ <http://en.wikibooks.org/wiki/Ubuntu>

Eventually you'll also need a 2D image editor. **GIMP** ([download](#)⁴) or Pinta⁵ (based on Paint .NET⁶) are highly-capable open-source editors, but proprietary commercial software such as Photoshop or Corel could also serve.

1.0.3. Where to Go for Help

In case you get stuck, you can ask for help from other Blender users⁷ in the appendices⁸.

1.1. Additional Resources

Many modules have a section like this at the bottom, listing websites with information on the topics covered in the module.

- Blender system requirements⁹
- the *GIMP*¹⁰ wikibook
- Windows installation¹¹
- Linux installation¹²
- OSX installation¹³
- About the Python programming language¹⁴

4 <http://www.gimp.org/downloads/>

5 <http://en.wikipedia.org/wiki/Pinta%20%28software%29>

6 <http://en.wikipedia.org/wiki/Paint%20.net>

7 [Chapter 196 on page 1425](#)

8 <http://en.wikibooks.org/wiki/..%2F%23Appendices>

9 <http://www.blender.org/features-gallery/requirements/>

10 <http://en.wikibooks.org/wiki/GIMP>

11 http://wiki.blender.org/index.php/Doc:Manual/Introduction/Installing_Blender/Windows

12 http://wiki.blender.org/index.php/Doc:Manual/Introduction/Installing_Blender/Linux

13 http://wiki.blender.org/index.php/Doc:Manual/Introduction/Installing_Blender/Mac

14 <http://www.python.org/about/>

2. What Blender Can Do

In this module, you'll learn what Blender does, both in terms of the product (images) and the process (3D modeling).

Blender is a free software¹ package for authoring "three-dimensional" (3D) graphics, including still images, games, and video.

While the end-product of most Blender projects is a two-dimensional (2D) raster image on a flat surface (be it a monitor, movie screen, or sheet of paper), the images are said to be "3D" because they exhibit the **illusion of depth**. In other words, someone looking at the image can easily tell which parts are meant to be closer and which are farther away.

2.0.1. An Example

Here's a realistic still image that was authored with Blender.



Figure 1 "A Lonely House", by Mayqel

Look closely at the building.

¹ <http://en.wikipedia.org/wiki/free%20software>

- Because it obscures the tree-lined hillside, you can tell that the hillside is *behind* the building instead of vice versa.
- The way the top and bottom edges of the front wall appear to converge toward the base of the tree allow you to judge the angle between the front wall and your viewpoint.
- Your brain interprets dark portions of the wall as shadows, allowing you to estimate where the light is coming from, even though the sun is outside the frame of the image.

While an illusion of depth can be authored by hand with 2D graphics software (or a paintbrush!), Blender provides a much easier way.

It's likely that the lonely house never existed outside of the artist's mind. Instead of building a big set on a rural lot in Germany, waiting for the right light, and photographing it, the author built a scene in a virtual 3D world—one contained inside a computer. He or she then used Blender to **render** the scene (convert it into a 2D image). You can view more of what Blender can do at the Blender gallery: <http://www.blender.org/features-gallery/>

2.0.2. Steps in the Modeling Process

In 3D computer graphics, a **model** is an abstract version of an object or scene that is suitable for rendering. **Modeling** is the process of making models.

Authoring the 3D scene for “A Lonely House” probably involved:

1. **Object modeling** — describing the shapes of objects, such as birds and trees, in three dimensions
2. **Shading** — applying patterns of color and texture to each object
3. **Lighting** — describing the intensity, color, direction and so on of the light source

If “A Lonely House” were made into a video, there would be additional steps of **animating** the scene (describing how it changes over time):

1. **Rigging** — describing how parts of a creature, such as the birds' wings, can move relative to other parts
2. **Posing** — choreographing the positions of the objects and their parts in the 3D scene over time

Blender is a capable tool for each and all of these steps.

2.1. Additional Resources

- w:3D rendering²
- w:Comparison of 3D computer graphics software³
- w:Computer-generated imagery⁴
- w:Depth perception⁵

2 <http://en.wikipedia.org/wiki/3D%20rendering>

3 <http://en.wikipedia.org/wiki/Comparison%20of%203D%20computer%20graphics%20software>

4 <http://en.wikipedia.org/wiki/Computer-generated%20imagery>

5 <http://en.wikipedia.org/wiki/Depth%20perception>

- Blender Art Gallery⁶
- Blender Homepage⁷

⁶ <http://www.blender.org/features-gallery/gallery/art-gallery/>
⁷ <http://www.blender.org/>

3. 3D Geometry

If you haven't previously studied 3D graphics, technical drawing, or analytic geometry, you are about to learn a new way of visualizing the world, an ability that's fundamental to working with Blender or any 3D modeling tool.

3D modeling is based on **geometry**, the branch of mathematics concerned with spatial relationships, specifically **analytical geometry**, which expresses these relationships in terms of algebraic formulas. You may have studied geometry in school, in which case, some of the terminology will seem familiar.

3.1. Coordinates And Coordinate Systems

Look around the weird room you're in. The odds are it will have a cuboidal shape, with four vertical walls at right angles to each other, a flat, horizontal floor, and a flat, horizontal ceiling.

Now imagine there's a scary wasp buzzing around the room. If there really is a scary wasp buzzing around, feel free to think about something else. Whatever it is, it's moving in three-dimensional space. In mathematical terms, what that means is that, at any moment, its position within the room can be expressed in terms of three unique numbers.

Now, there are an infinite number of ways—*coordinate systems*—in which we could come up with a convention for defining and measuring these numbers—the *coordinates*. And each convention will give different values for the numbers, even if the fly is in the same position—coordinates only make sense with reference to a specific coordinate system! To narrow down the possibilities (in a purely arbitrary fashion, of course), let us label the walls of the room with the points of the compass: in a clockwise direction, North, East, South and West. (If you know which way is really north, feel free to use that to label the walls of your room; otherwise, choose any wall you like as north.)

Consider the south-west corner of your room, the point at floor level. We will call this (arbitrary) point the *origin* of our coordinate system, and the three numbers at this point will be $(0, 0, 0)$. The first of the three numbers will be the distance (in some suitable units, let's say meters) eastwards from the west wall, the second number will be the distance northwards from the south wall, and the third number will be the height above the floor.

Each of these directions is called an *axis* (plural: *axes*), and they are conventionally labelled X (the horizontal width or breadth), Y (the horizontal length) and Z (the vertical height), in that order. With a little bit of thought, you should be able to convince yourself that every point within the space of your room corresponds to exactly one set of (x, y, z) values. And conversely that every possible combination of (x, y, z) values, with $(0 \leq x \leq W)$, $(0 \leq y \leq L)$ and $(0 \leq z \leq H)$ where W is the east-west dimension of your room, L is its north-south dimension, and H is the height between ceiling and floor) corresponds to a point in the room.

The following diagram illustrates how the coordinates are built up, using the same colour codes that Blender uses to label its axes: red for X, green for Y and blue for Z. In the second picture, the value for x defines a plane parallel to the west wall of the room; in the third picture, the value for y defines a plane parallel to the south wall; and in the fourth picture, the value for z defines a plane parallel to the floor. Put the planes together in the fifth picture, and they intersect at a unique point.



Figure 2

This style of coordinate system, with the numbers corresponding to distances along perpendicular axes, is called *cartesian coordinates*, named after René Descartes, the 17th-century mathematician who first introduced the concept. Legend has it that he came up with the idea after watching a fly buzzing around his bedroom! Well, people of old were observant, no wonder!

There are other ways to define coordinate systems, for example by substituting direction angles in place of one or two of the distance measurements. These can be useful in certain situations, but usually all coordinate systems in Blender are cartesian. However, switching between these dimensions are simple and easy to do though in Blender.

3.1.1. Negative Coordinates

Can coordinate values be negative? Depending on the situation, yes. Here we are only considering points within our room. But supposing, instead of placing our origin in the bottom southwest corner, we put it in the middle of the room, halfway between the floor and ceiling. (After all, it is an arbitrary point, we can place it where we like, just so long as we all agreed on where it is.) Now, remember that our X-coordinate is the distance *east* from the origin; How do we express the fact that a point lies *west* of the origin? Simple: We give it a negative X-coordinate. Similarly, points north of the origin have a positive Y-coordinate, while those south of it get a negative one; and those points above the origin get a positive Z-coordinate, while those below get a negative one.

3.1.2. Handedness Of Coordinate Systems

It is conventional for most cartesian coordinate systems to be *right-handed*. To see what this means, hold the thumb, index finger and middle finger of your right hand perpendicular to each other:

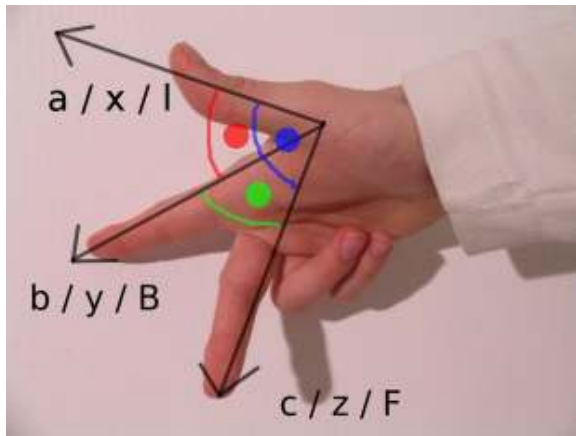


Figure 3 The three axes form a right-handed system

Now you can orient your hand so that the thumb points along the X-axis in the positive direction (direction of increasing coordinate numbers), the index finger along the positive Y-axis, and the middle finger along the positive Z-axis. Another way of looking at it is, if you placed your eye at the origin, and you could see the three arrows pointing in the direction of positive X, positive Y and positive Z as in Figure 1 below, the order X, Y, Z would go clockwise.

3.2. Axes Of Rotation

Consider a spinning ball. Every point on it is moving, except the ones right in the middle: these form a line of stillness around which the rest of the ball spins. This line is called the *axis of rotation*.

More precisely, the axis of rotation of a rotating object over a period of time is a point or a line connecting points that do not change position while that object rotates, drawn when the observer assumes he/she does not change position relative to that object over time.

Conventionally, the *direction* of the axis of rotation is such that, if you place your eye looking in that direction, the rotation appears clockwise, as illustrated below, where the yellow arrow shows the rotational movement, while the purple one shows the rotation axis:



Figure 4

To remember this convention, hold your right hand in a thumbs-up gesture:



Figure 5

If the rotation follows the direction of the curled-up fingers, then the direction of the axis of rotation is considered to be the same as the thumb is pointing.

Therefore, when describing the direction of a rotating object, do not say it rotates left to right, right to left, clockwise or counterclockwise. All of the above is relative to the observer. Instead of saying this, find the direction of the axis of rotation and draw an arrow to represent it.

3.3. Additional Resources

- the *Geometry*¹ wikibook

¹ <http://en.wikibooks.org/wiki/Geometry>

- w:Analytic geometry²
- w:Cartesian coordinate system³
- w:Right-hand rule⁴
- w:Rotation⁵

2 <http://en.wikipedia.org/wiki/Analytic%20geometry>

3 <http://en.wikipedia.org/wiki/Cartesian%20coordinate%20system>

4 <http://en.wikipedia.org/wiki/Right-hand%20rule>

5 <http://en.wikipedia.org/wiki/Rotation>

4. Coordinate Transformations

4.1. Coordinate Transformations

A *transformation* is any kind of operation that changes coordinate values in some way. For example, if you pick up an object and move it to a different place in the room without changing its orientation, then the coordinates of each point on the object relative to the room are adjusted by an amount that depends on the distance and direction between the old and new positions; this is called a *translation* transformation.

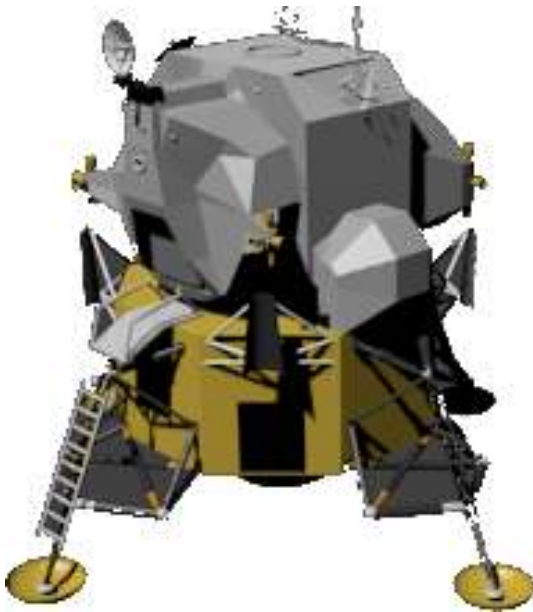


Figure 6 Object at original location

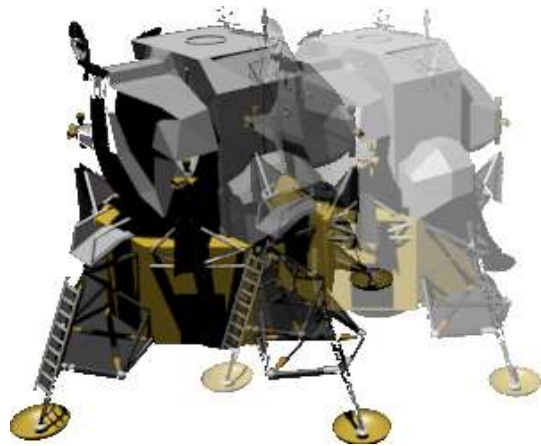


Figure 7 Object translated to new position

If you simply turn the object around without moving it from its original location, then that's called a *rotation*.

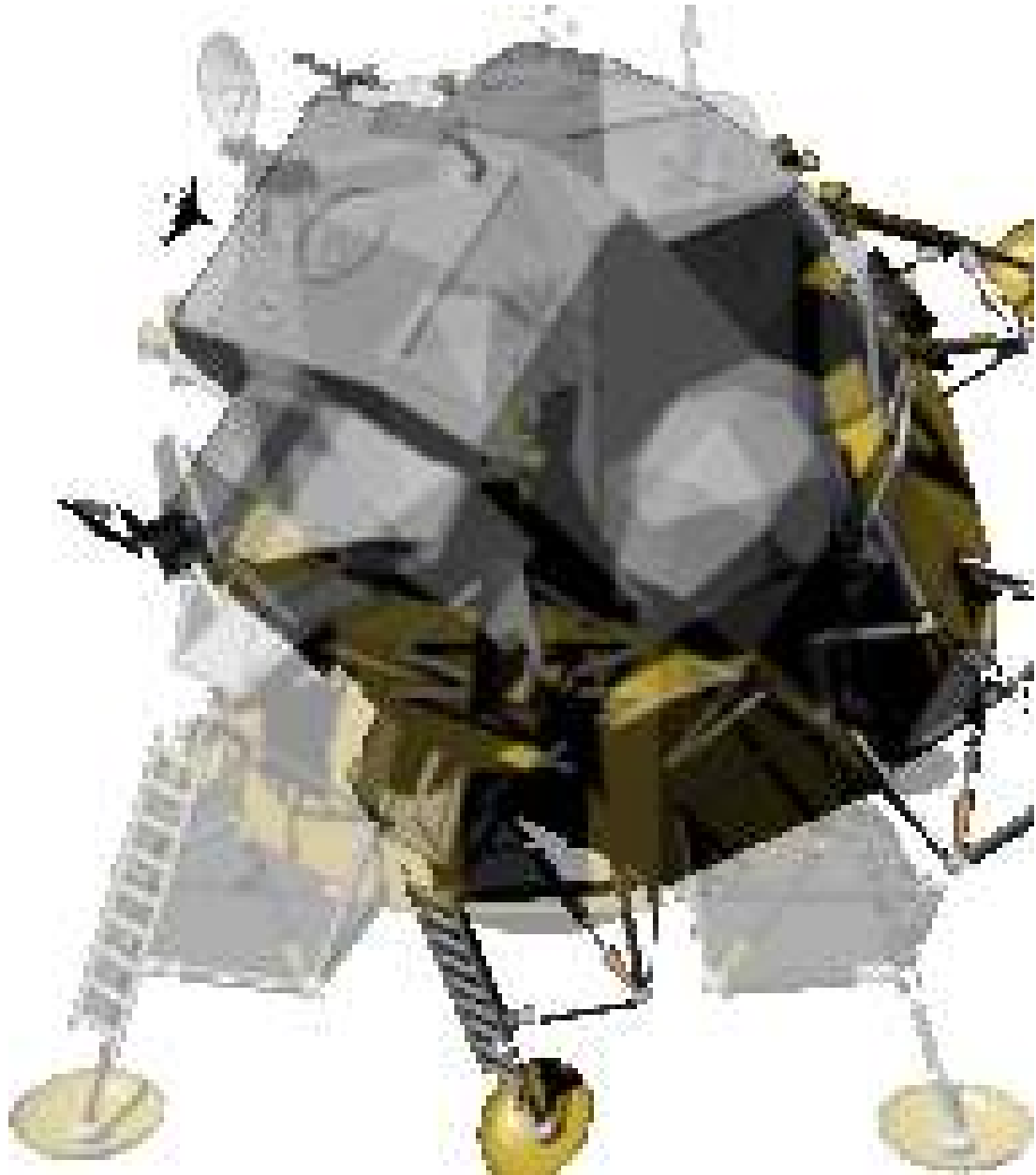


Figure 8 Object rotated 45°

If the object were to get bigger or smaller, then that would be called a *scaling* transformation. In the real world, only a few objects can be scaled in this way—for example, a balloon can be inflated or deflated to a larger or smaller size, but a bowling ball cannot be so resized. But in the world of computer graphics, any object can be scaled. Scaling may be *uniform* (applying equally in all dimensions) or *non-uniform*.



Figure 9 Object uniformly scaled to 50% of original size

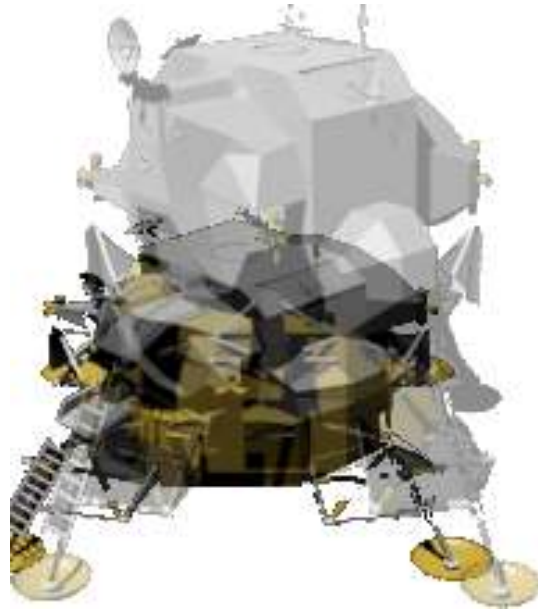


Figure 10 Object scaled vertically to 50% of original size

4.2. Linear Transformations

The main kinds of coordinate transformations we're concerned with here are called *linear transformations*. Lines that were straight before the transformation remain straight after; they do not turn into curves. For example, the following diagram illustrates three linear transformations applied to the square in the center: a scaling, a *shear* or *skew* transform, and a rotation, plus one transformation that isn't linear, because it introduces curves where there were none before.

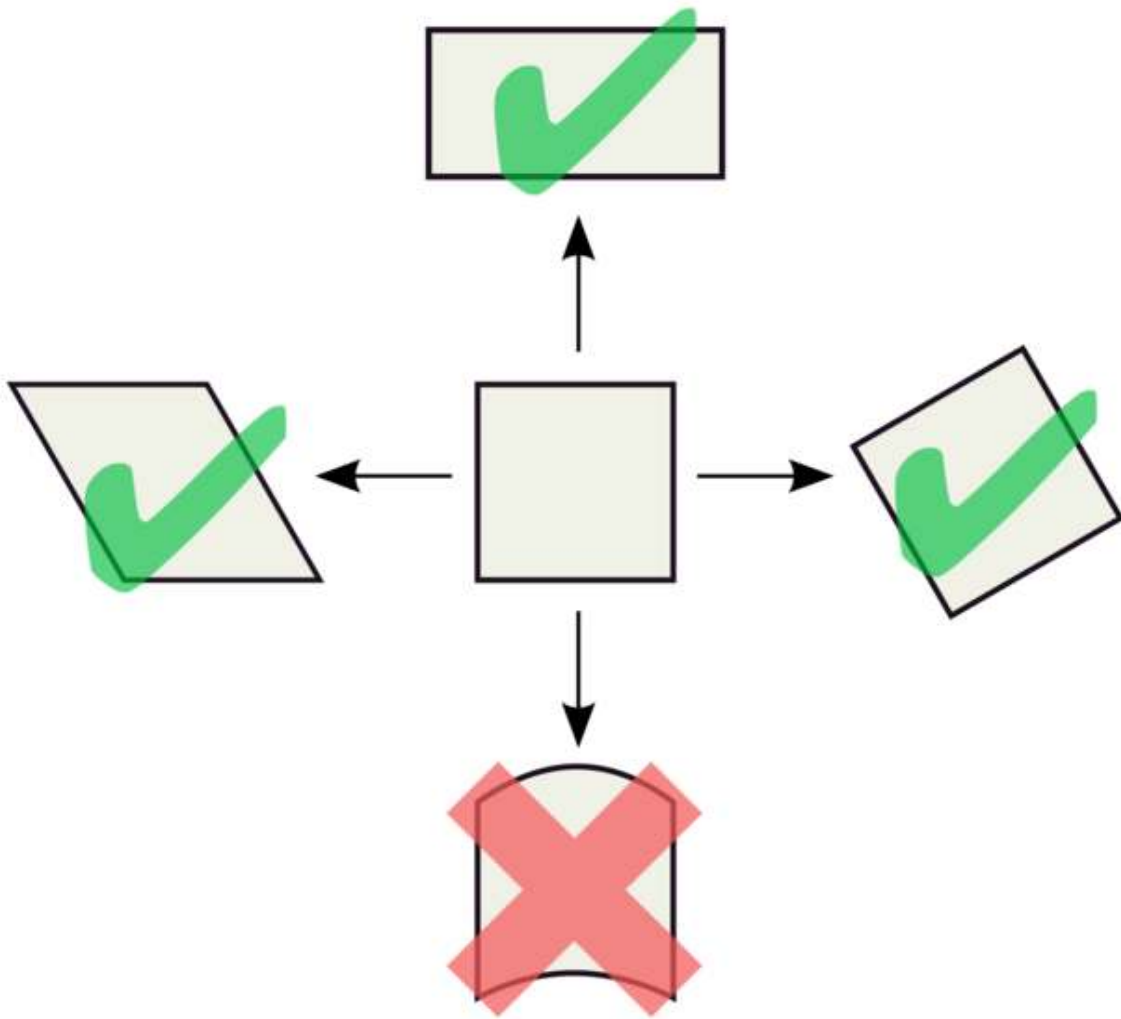


Figure 11

4.3. Multiple Transformations

It is possible to *concatenate* or *compose* a series of transformations. The resulting transformation can do a whole bunch of things in one operation—translation, rotation, scaling etc. However, the order of composition of the component transformations becomes important—in general, transformations are not *commutative*. For example, compare the result of moving our model a certain distance along the Y axis followed by rotating it about the X axis:

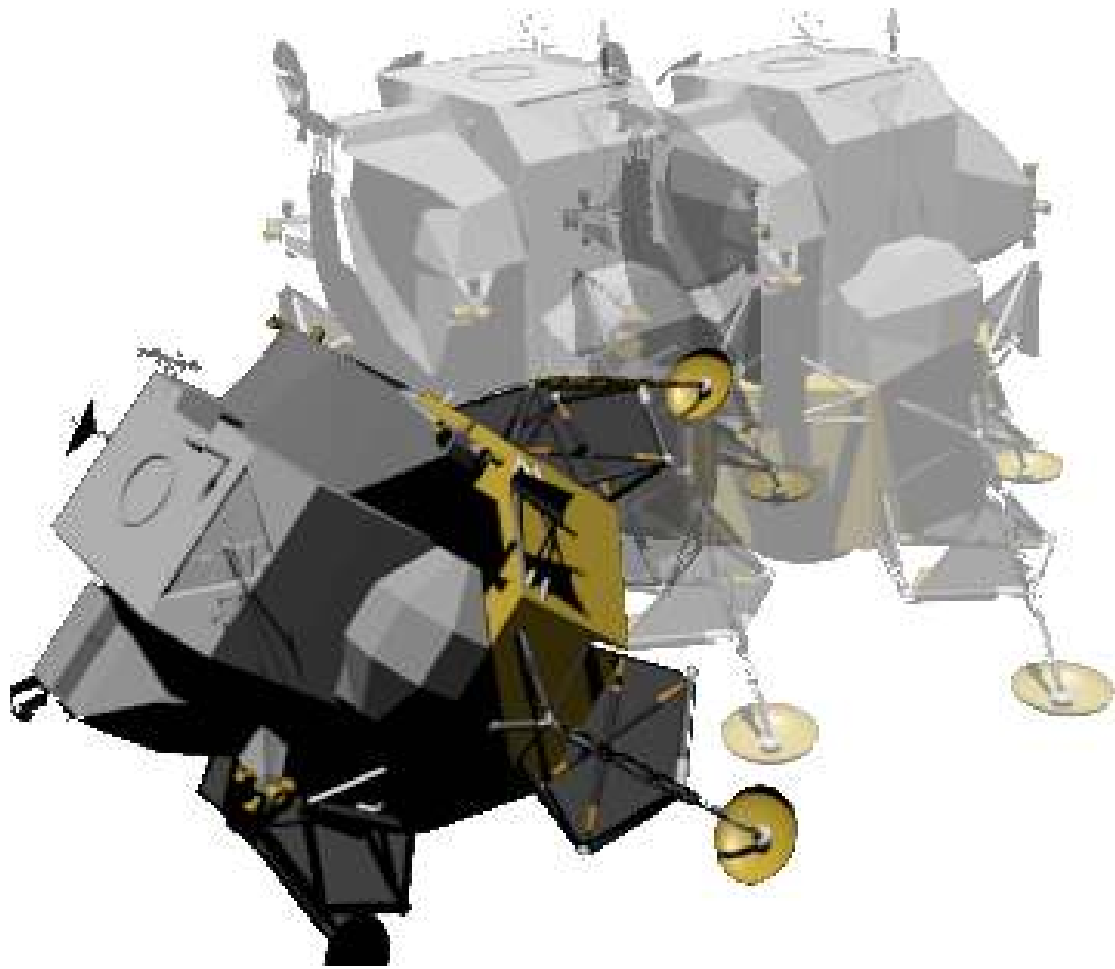


Figure 12 Translation followed by rotation

versus the result of doing the rotation first:

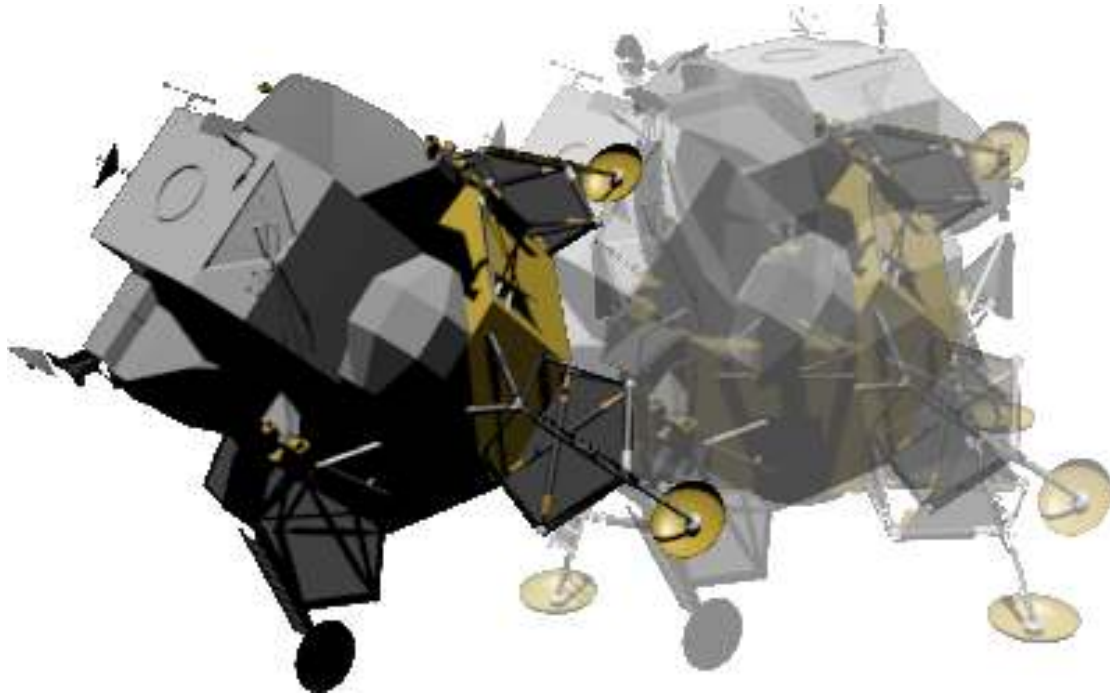


Figure 13 Rotation followed by translation

In some instances, three of the forms of transformation may be applied on a single object concurrently. Such a feature exists in Blender and is normally implemented in creating animations. Therefore, you can decide to pick up the object (first transformation - location), twist it (second transformation - rotation), and, in a 3D modeling environment, increase the size of the object (third transformation - scaling).

4.4. Inverse Transformations

There is quite commonly a need to find the *inverse* of a transformation—that is, the opposite transformation that undoes its effect. For example, a rotation of $+45^\circ$ about the X axis is undone by a rotation of -45° around the same axis.

Inverses have many uses, one of which is to simplify the construction of certain kinds of transformations.

For example, it is easy to construct a rotation transformation about the X, Y or Z-axis of the coordinate system. But what about a rotation of Θ° around an arbitrary axis? This can be made out of the following parts:

- a translation that makes the rotation axis pass through the origin.
- rotations about the Y and/or Z axes, as appropriate, so the rotation axis lies along the X axis.
- a rotation of Θ° about the X axis.
- the inverse of the rotations that aligned the rotation axis with the X axis.
- the inverse of the translation that made the rotation axis pass through the origin.

Most of the transformations we deal with in 3D modelling have an inverse, but not all—see the next section for some that don't.

4.5. Projections

Most of our display and output devices are not three-dimensional. Thus, three-dimensional images need to be *projected* onto a two-dimensional surface (like a display screen or a printed page) before we can see them.

There are two main ways to perform such projections. One is *orthographic* projection, where parallel lines are drawn from all points of the three-dimensional object until they intersect a plane representing the display surface:

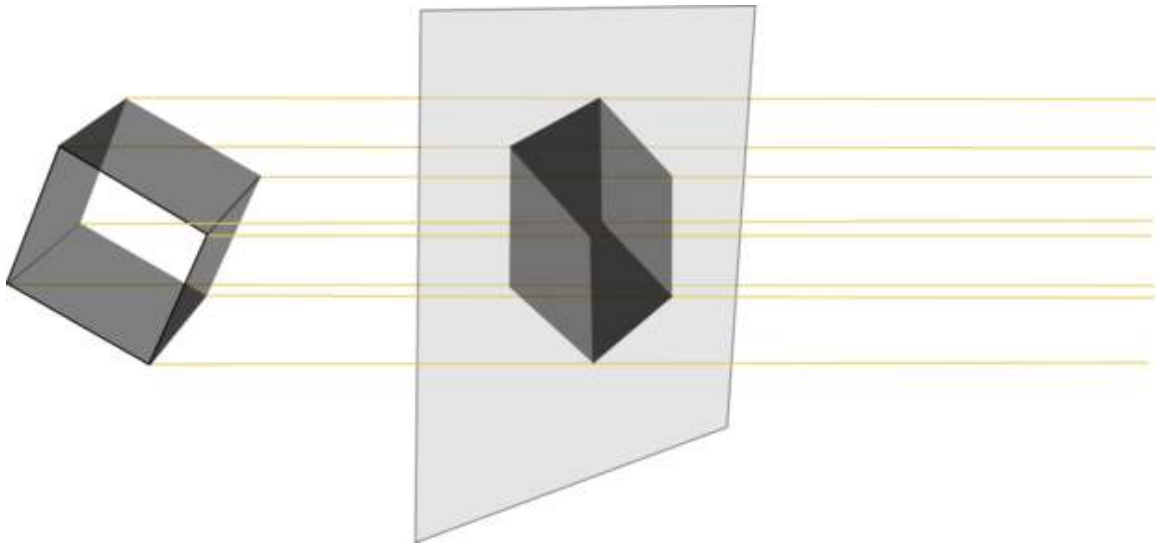


Figure 14

More on orthographic projections¹

The other way is *perspective* projection, where the lines drawn are not parallel, but intersect at a point representing the location of the eye of the viewer:

¹ Chapter 5 on page 27

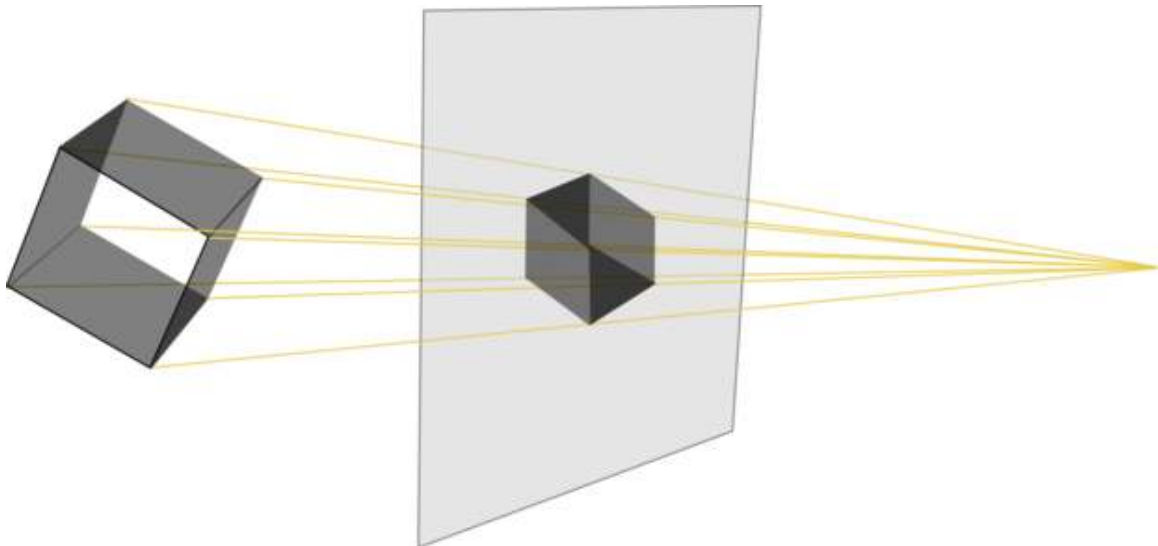


Figure 15

Projections are also linear transformations. But since they take a three-dimensional space and flatten it onto a two-dimensional surface, some information is lost. Which means these are *non-invertible* transformations; they cannot be undone, at least in a unique way—the depth information is gone.

More on perspective projections²

The mathematics of perspective were first worked out in the 11th century by Alhazen³, and used to great effect by the Italian Renaissance painters four hundred years later.

--Nkansahrexford⁴ (discuss⁵ • contribs⁶) 01:15, 11 February 2012 (UTC)Nkansahrexford

2 Chapter 6 on page 31

3 <http://en.wikipedia.org/wiki/Alhazen>

4 <http://en.wikibooks.org/wiki/User%3ANkansahrexford>

5 <http://en.wikibooks.org/wiki/User%20talk%3ANkansahrexford>

6 <http://en.wikibooks.org/wiki/Special%3AContributions%2FNkansahrexford>

5. Orthographic Views

Blender provides two different ways of viewing 3D scenes:

- orthographic view

and

- perspective view.

In order to use these views effectively, you need to understand their properties.

An **orthographic** view (or projection) of a 3D scene is a 2D picture of it in which parallel lines appear parallel. An additional property of such views is that all edges perpendicular to the view direction appear in proportion, at exactly the same scale.

Usually such views are aligned with the scene's primary axes. Edges parallel to the view axis disappear; those parallel to the other primary axes appear horizontal or vertical. The commonly-used orthographic views are the front, side, and top views, though back and bottom views are possible.

Uniform scale makes orthographic views very useful when constructing 3D objects—not only in computer graphics, but also in manufacturing and architecture.

Here's one way to think about orthographic views:

Imagine photographing a small 3D object through a telescope from a very great distance. There would be no foreshortening. All features would be at essentially the same scale, regardless of whether they were on the near side of the object or its far side. Given two (or preferably three) such views, along different axes, you could get an accurate idea of the shape of the object—handy for "getting the feel" of objects in a virtual 3D world where you're unable to touch or handle anything!

5.1. Example

Here is a drawing of a staircase:

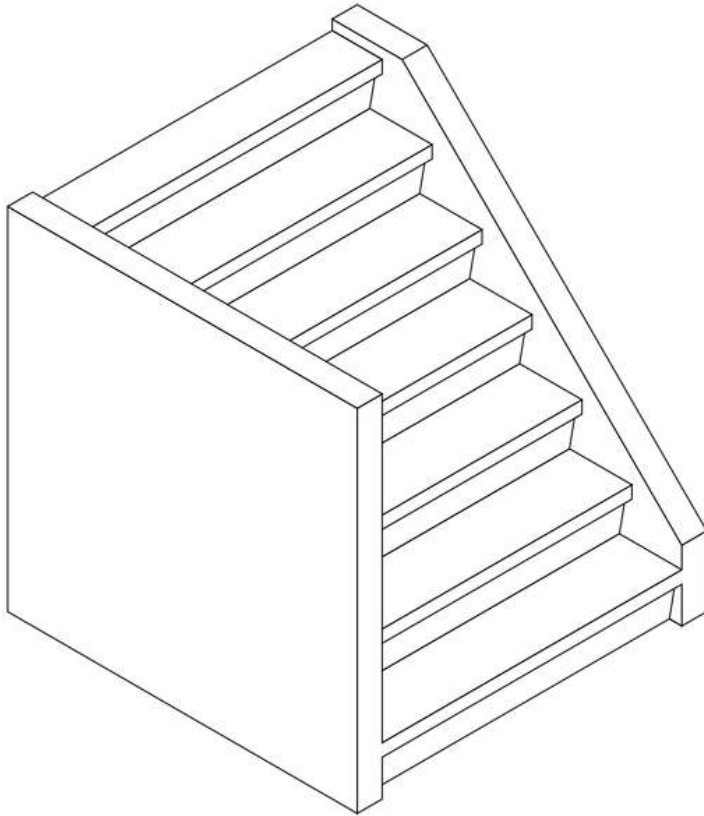


Figure 16 An isometric view of a staircase

and here are three orthographic views of the same staircase, each outlined in red:

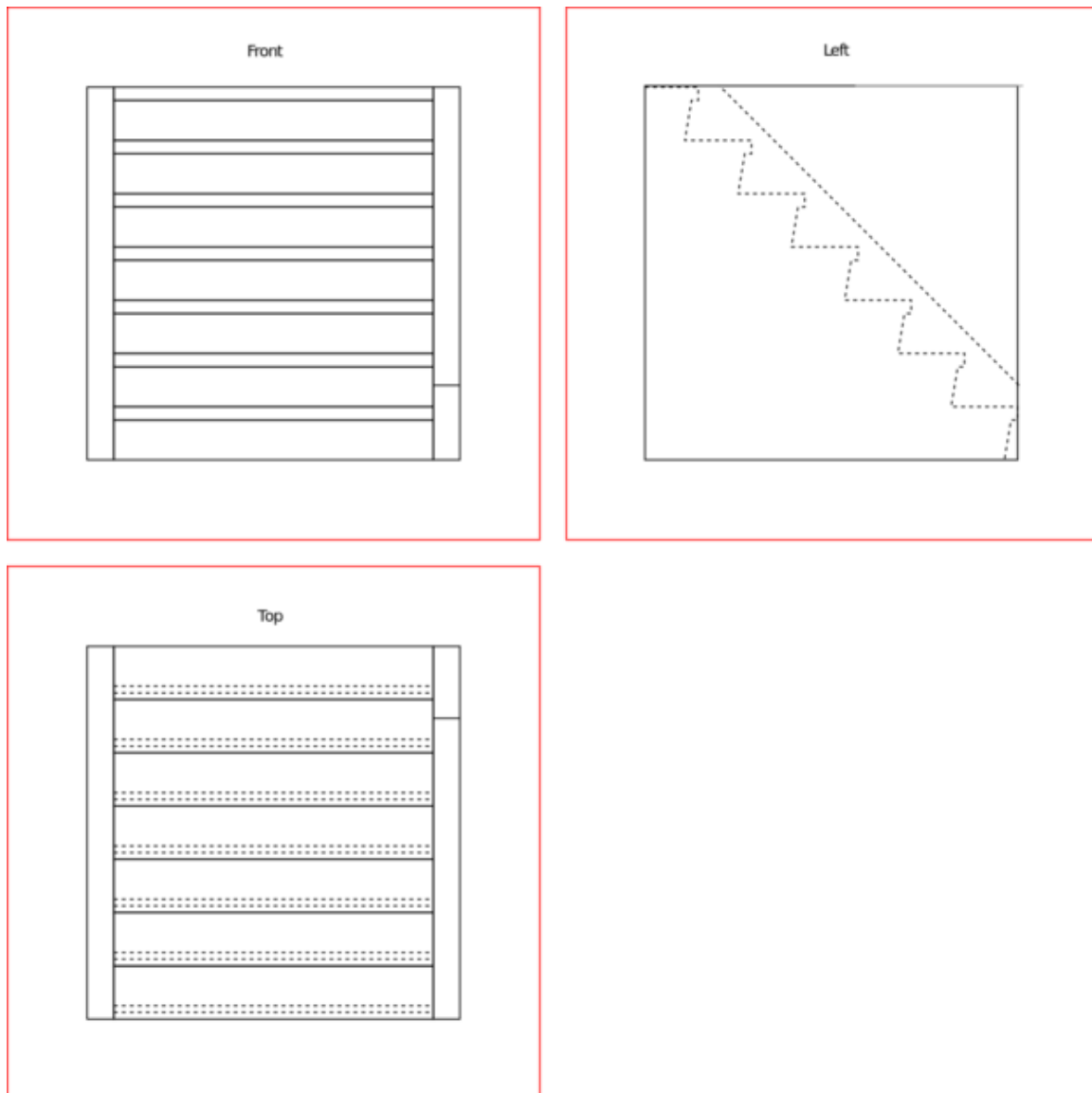


Figure 17 Figure 1: Orthographic views of a staircase

The views are from the front, top, and left. Dashed lines represent edges that, in real life, would be hidden behind something, such as the left wall of the staircase. (Think of each view as an X-ray image.)

The leading edges of the steps are visible in both the front and top views. Note that they appear parallel and of equal length in 2D, just as they are in 3D reality.

5.2. Additional Resources

- [w:Orthographic projection](#)¹

¹ <http://en.wikipedia.org/wiki/Orthographic%20projection>

6. Perspective Views

As you know, the main reason for modeling 3D objects in Blender is to render images that exhibit the illusion of depth.

Orthographic views are great for building a house, but seriously flawed when it comes to creating realistic images of the house for use in a sales brochure. While a builder wants blueprints that are clear and accurate, a seller wants imagery that's aesthetically pleasing, with the illusion of depth. Blender makes it easy to use tricks like perspective, surface hiding, shading, and animation to achieve this illusion.

How does perspective work?

The essence of **perspective** is to represent parallel edges (in a 3D scene) by edges (in the 2D image) that *are not parallel*. When done correctly, this produces **foreshortening** (nearby objects depicted at a larger scale than faraway ones) and contributes to the illusion of depth.

Perspective is challenging to draw by hand, but Blender does it for you, provided you give it a 3D model of the scene and tell it where to view the scene from.

If you're confident that you understand perspective, you can skip the rest of this module and proceed to the "Coordinate Spaces in Blender" module .

Note:

Blender only does 3-point perspective, not 1-point or 2-point.

6.1. One-point Perspective

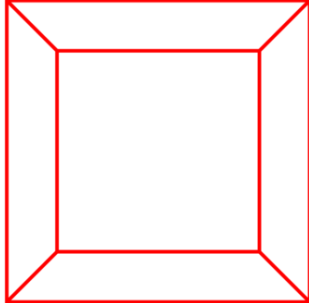


Figure 18 **Figure 1:** 1-Point Perspective.

Drawing classes teach several different kinds of perspective drawing: one-point perspective, two-point perspective, and three-point perspective. In this context, the word "point" refers to what artists call a *vanishing point*.

One-point perspective works when you're looking at a 3D object head-on and it's centered in your view.

Imagine looking down a straight and level set of train tracks. The tracks will appear to converge at a point on the horizon. This is the **vanishing point**.

The image on the right is a 2D image of a cubic lattice or framework. Like any cube, it has six square faces and twelve straight edges. In the 3D world, four of the edges are parallel to our line-of-sight; they connect the four corners of the nearest square to the corresponding corners of the farthest one. Each of these edges is parallel to the other three.

In the 2D image, those same four edges appear to converge toward a vanishing point, contributing to the illusion of a depth. Since this is one-point perspective, there is a single point of convergence at the center of the image.

6.2. Two-point Perspective

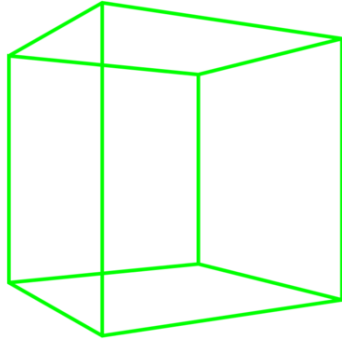


Figure 19 **Figure 2:** 2-Point Perspective.

Now the cube is at eye level, and you're near one of its edges. Since you're not viewing it face-on, you can't draw it realistically using one-point perspective. The horizontal edges on your left appear to converge at a point on the horizon to the left of the cube, while those on the right converge to the right. To illustrate the cube with a good illusion of depth, you need two vanishing points.

6.3. Three-point Perspective

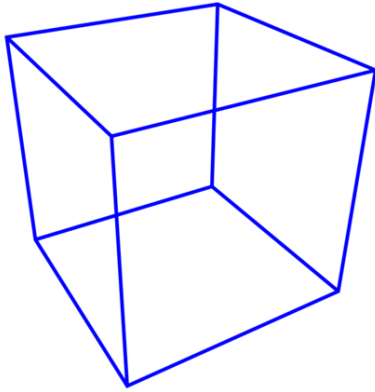


Figure 20 3-Point Perspective.

Now imagine that you're above the cube near one of its corners. To draw this, you'd need three vanishing points, one for each set of parallel edges.

From this perspective, there are no longer any edges which appear parallel. The four vertical edges, the four left-right edges, and the four in-out edges each converge toward a different vanishing point.

6.4. Additional Resources

- [w:Perspective \(graphical\)](#)¹

¹ <http://en.wikipedia.org/wiki/Perspective%20%28graphical%29>

7. Coordinate Spaces in Blender

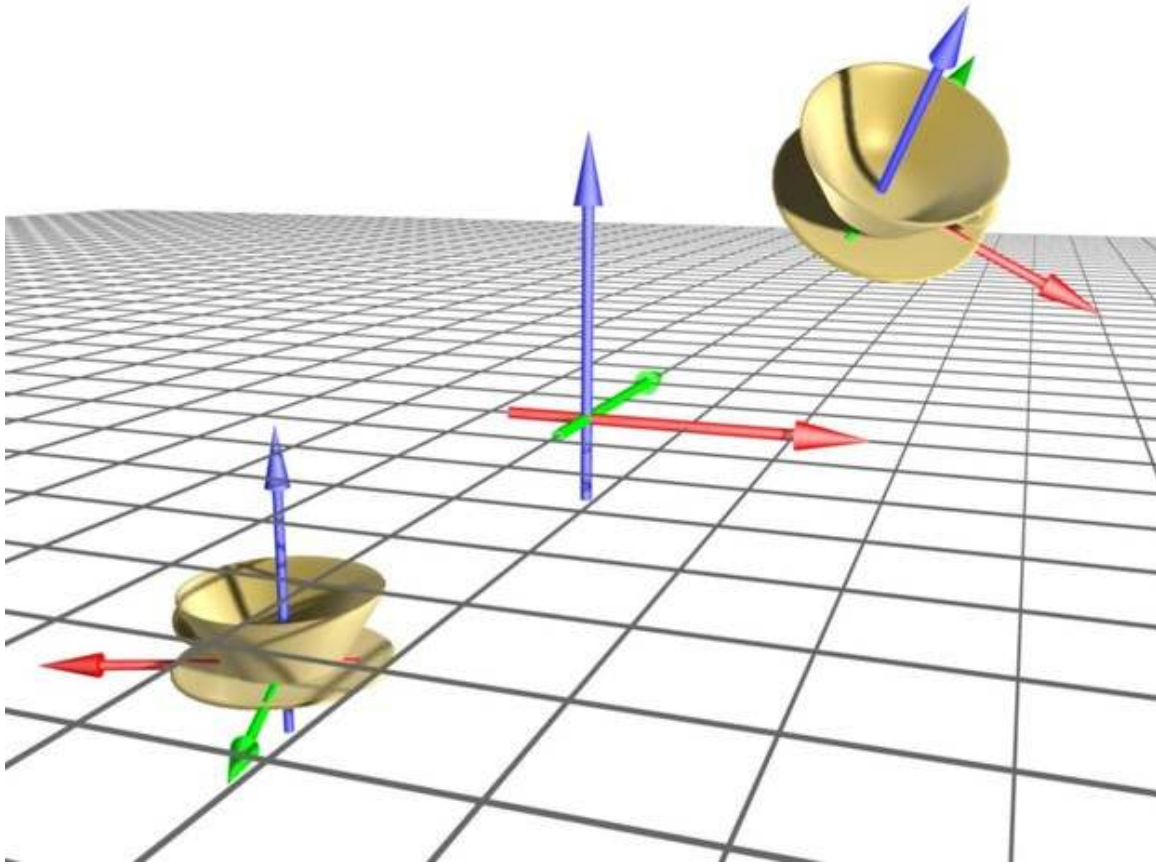


Figure 21 Figure 1: Objects in a three dimensional space. In the center of the coordinate system is the origin of the global coordinate system.

We'll start looking at how 3D scenes are represented in Blender.

As was explained in the "3D Geometry" module, Blender represents locations in a scene by their coordinates. The coordinates of a location consist of three numbers that define its distance and direction from a fixed origin. More precisely:

- The first (or x-) coordinate of the location is defined as its distance from the YZ plane (the one containing both the Y and Z axes). Locations on the +X side of this plane are assigned positive x-coordinates, and those on the -X side are given negative ones.
- Its second (or y-) coordinate is its distance from the XZ plane, with locations on the -Y side of this plane having negative y-coordinates.
- Its third (or z-) coordinate is its distance from the XY plane, with locations on the -Z side of this plane having negative z-coordinates.

Thus the origin (which lies at the junction of all three axes and all three planes) has coordinates (0, 0, 0).

7.1. Global and local coordinates

Blender refers to the coordinate system described above as the **global coordinate system**, though it's not truly global—each scene has its own global coordinate system. Each global coordinate system has a fixed origin and a fixed orientation, but we can view it from different angles by moving a virtual camera through the scene and/or rotating the camera.

Global coordinates are adequate for scenes containing a single fixed object and scenes in which each object is merely a single point in the scene. When dealing with objects that move around (or multiple objects with sizes and shapes), it's helpful to define a **local coordinate system** for each object, a coordinate system that can move with the object and follow the object around. The origin of an object's local coordinate system is often called the **center of the object** even though it needn't coincide with the geometrical center of the object.

3D objects in Blender are largely described using **vertices** (points in the object, singular form: **vertex**). The global coordinates of a vertex depend on:

- the (x, y, z) coordinates of the vertex in the object's **local** coordinate system
- the location of the object's center
- any rotation (turning) of the local coordinates system relative to the global coordinate system, and
- any scaling (magnification or reduction) of the local coordinate system relative to the global coordinate system.

For example, the teacup in Figure 1 is described by a mesh model containing 171 vertices, each having a different set of local (x, y, z) coordinates relative to the cup's center. If you **translate** the cup (move it without rotating), the only bits of the model that have to change are the global coordinates of the center; the local coordinates of all its vertices would remain the same.

7.1.1. Coordinates of child objects

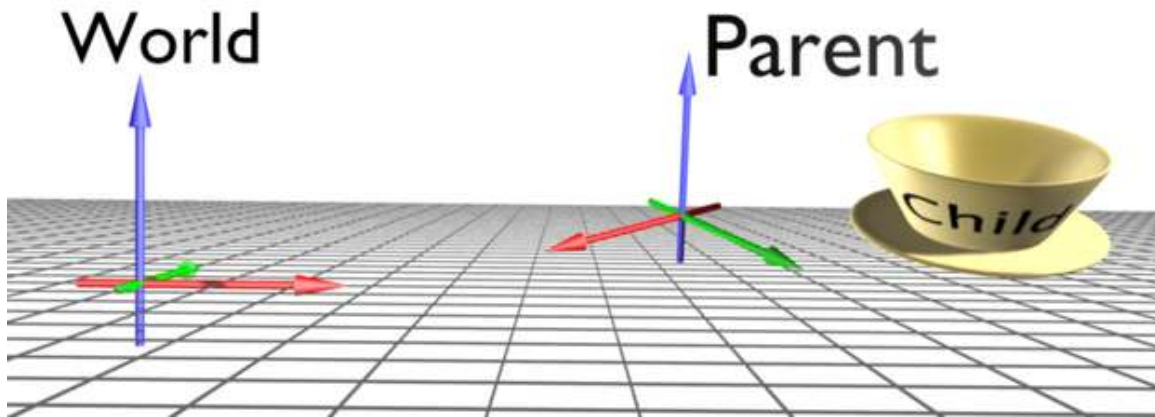


Figure 22 Figure 1b: A parent serves as the source of the global coordinates for its child object. The child is the cup; the parent's orientation is shown with the colored arrows.

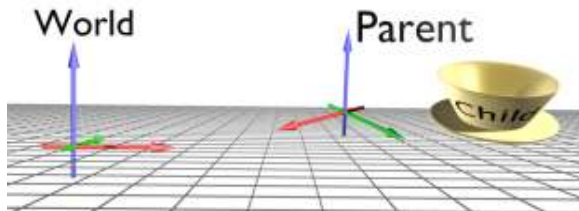


Figure 22 Animation of the above

Any object can act as a **parent** for one or more other objects in the same scene, which are then referred to as its **children**. (An object cannot have more than one direct parent, but parent objects may themselves be the children of other objects.)

If an object has a parent, its position, rotation, and scaling are measured in the parent's local coordinate system, almost as if it were a vertex of the parent. That is to say, the position of the child's center is measured from the parent's center instead of the origin of the global coordinate system. So if you move a parent object, its children move too, even though the children's coordinates have not changed. The orientation and scaling of a child's local coordinate system are likewise measured relative to those of its parent. If you rotate the parent, the child will rotate (and perhaps revolve) around the same axis.

Parent-child relationships between objects make it simpler to perform (and animate) rotations, scaling and moving in arbitrary directions. In **Fig. 1b** the teacup is a child object of the coordinate cross on the right. That cross is itself the child of an invisible parent. (It is both a parent and child.) In the cup's local coordinate system, it is not rotating, but as the cross on the right rotates around its Z axis, it causes the cup to rotate and revolve. In real animations, it will be much easier when the character holding the cup rotates, the cup changes its position respectively.

7.2. View coordinates

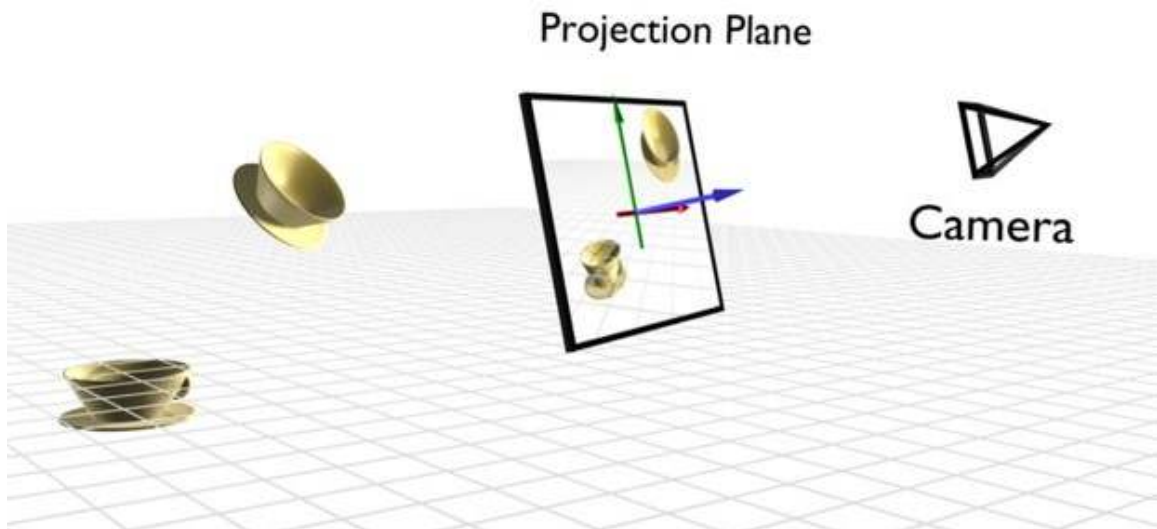


Figure 23 **Figure 2:** View coordinates and Projection Plane

Taking the viewer of the scene in consideration, there is another coordinate space: the view coordinates. In **Fig. 2** the viewer is symbolized by the camera. The Z axis of the view coordinates always points directly to the viewer in orthographic projection. The X axis points to the right, the Y axis points upwards (**Fig. 3**).

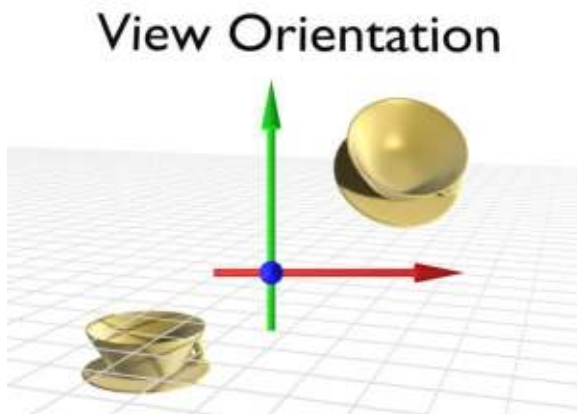


Figure 24 **Figure 3:** View coordinates in viewing direction

In fact you always work in view coordinates if you don't set it any other way*. This is particularly handy if you have aligned your view prior to modeling something, e.g. if an object has a slanted roof and you want to create a window to fit in this roof, it would be very complicated to build this window aligned to the local coordinate system of the object, but if you first align your view to the slanted roof you can easily work in this view coordinate system.

(* In the Blender 2.6 series, the default has been changed to global coordinates. View coordinates remain as an option.)

If you work in one of the three standard views (Front/Top/Side) the alignment of the view coordinates fits the global coordinates; therefore it is quite natural to model in one of the standard views and many people find this the best way to model.

7.3. Normal coordinates

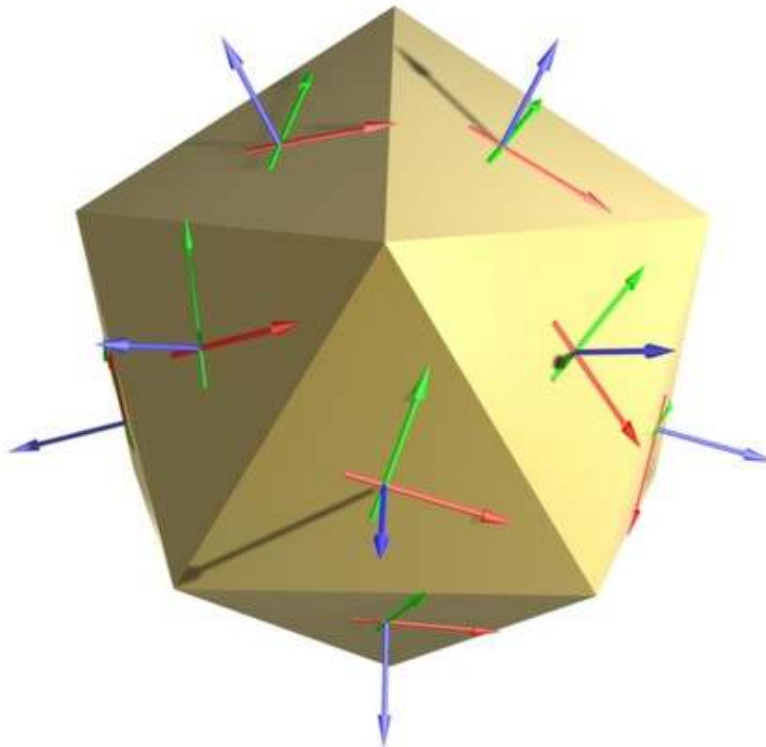


Figure 25 **Figure 4:** Normal coordinate spaces for faces. The normal is shown in blue.

Although Blender is a 3D program, only the faces are visible. The orientation of the faces is important for many reasons. For example, in our daily lives it seems quite obvious that a book lies flat on a table. This requires the surface of the table and that of the book to be parallel to each other. If we put a book on a table in a 3D program, there is no mechanism that forces these surfaces to be parallel; we have to ensure that ourselves.

The orientation of a face can be described with the help of a so-called **surface normal**. It is always perpendicular to the surface. If several faces are selected, the resulting normal is averaged from the normals of every single face. In **Fig. 4** the normal coordinates of the visible faces are drawn.

This concept can be applied to individual points on the object, even if the points themselves have no orientation. The normal of a point is the average of normals of the adjacent faces.

The images for this tutorial were produced with Blender v2.46.

8. User Interface Overview

Blender's **user interface** (the means by which you control the software) is not particularly easy to learn. However, it has improved over time and is expected to continue doing so. The current version of the Blender software is 2.63. You can download it from the Blender website.

The tutorials in this section will familiarize you with the basics of the user interface. By the end of this section, you should be able to:

- resize, split, and merge any Blender window
- change the type of any Blender window
- access user preferences
- access panels containing buttons and other controls
- change the viewpoint of a viewport

For those new to Blender, this is the most important section of the book.

8.0.1. Advice on Customization

Blender is a complex software package with many customizable features. You can customize the user interface to assign new functions to buttons and hotkeys. In fact, you can change almost anything to suit yourself. However, this complicates the giving and following of directions. It is recommended that you adhere to the default screen arrangements of Blender in order to be able to follow well in the remaining parts of these tutorials. Blender ships along with some 4 to 5 screen-content arrangements which are suitable for almost any kind of job you'll want to use it for - from creating motions and animations to doing games.

We recommend leaving Blender's user interface in its "factory settings" while working through the *Noob to Pro* tutorials. At the very least, wait until you've mastered the basics before you customize the interface - and we know you definitely will when you master it!

Nkansahrexford¹ (discuss² • contribs³) 01:26, 11 February 2012 (UTC)Nkansahrexford

¹ <http://en.wikibooks.org/wiki/User%3ANkansahrexford>

² <http://en.wikibooks.org/wiki/User%20talk%3ANkansahrexford>

³ <http://en.wikibooks.org/wiki/Special%3AContributions%2FNkansahrexford>

9. Keystroke, Button, and Menu Notation

As you read through these tutorials, you will encounter cryptic codes such as `[SHIFT] + [LMB]` and *Timeline* → *End Frame*. These codes describe actions you perform using the keyboard and mouse. The notation used in this book comes from the Blender community, and it is standard there. We will try to import those standards here to facilitate our studies.

If you're reading this book online, you may wish to print this page for future reference. In addition, you can bookmark in your browser for faster reference when behind your PC.

9.1. Hotkeys



Figure 26 A typical numpad

Most computer keyboards have number keys in two different places: in a row above the letters and in a **numpad** (numeric keypad) on the right side of the keyboard. While many applications use these two sets of keys interchangeably, Blender does not. It assigns different functions to each set. If you're using a laptop keyboard without a separate numeric keypad, this might cause some difficulty. You'll need to use your *function key* to do some things. It is possible to indicate to Blender the type of keyboard you are using, but we strongly recommend you use a standard external keyboard even if you're on a laptop for these tutorials as it will make your studies and usage of Blender much more straightforward and enjoyable.

This book often assumes that your keyboard has a numpad. If it doesn't, consult the tutorial on Non-standard Input Devices¹ for alternative ways to access the numpad's functions.

¹ Chapter 10 on page 49

9.1.1. Notation





Notation	Action or Key
	(press and hold) the Alt (Option) key
	(press and hold) the Command or Super key (On a Windows keyboard, the key bearing the Windows logo; on a Macintosh, it bears the Apple logo.)
	(press and hold) the Ctrl (Control) key
	(press and hold) the Fn (Function) key (generally found only on laptops)
	(press and hold) the Shift key
	(press) the Enter (Return) key (on the main keypad)
	(press) the Esc (Escape) key
through	function keys F1 through F12 (often in a row along the top of the keyboard)
	(press) the space bar (usually unmarked)
	(press) the Tab key
through	the letters (on the main keypad)
through 	the numerals (above the letters) on the main keypad
through 	the numerals on the numpad
, , , , , , and 	other keys on the numpad
	(press) the Delete key (not !)
	(press) the down arrow key (not !)
	(press) the left arrow key (not !)
	(press) the right arrow key (not !)
	(press) the up arrow key (not !)





Combinations that involve holding down a key while performing another action are written with a plus sign (+). Thus:

- + means to while holding down
- and
- + + means to while holding down both and .

9.2. Mouse Notation

Blender uses three mouse buttons and the scroll wheel as well, if you have one. If your mouse only has one or two buttons, consult the tutorial on Non-standard Input Devices² for alternative ways to access the functions assigned to these buttons.

Notation	Action or Button
	click with the Left Mouse Button
	click with the Right Mouse Button
	press down on (don't turn) the scroll wheel <i>or</i> Middle Mouse Button
	turn the scroll wheel in either direction

Mouse and keyboard actions are often combined.  +  means to click  while holding down .

9.3. Navigating Menus

Blender uses both pop-up and pull-down/pull-up menus. Many menus have **submenus** (menus that are reached by going through another menu). If a menu item displays a black triangle, that means it leads to a submenu. For instance, the pull-down menu shown below has seven items, one of which ("Bake Render Meshes") leads to a submenu:

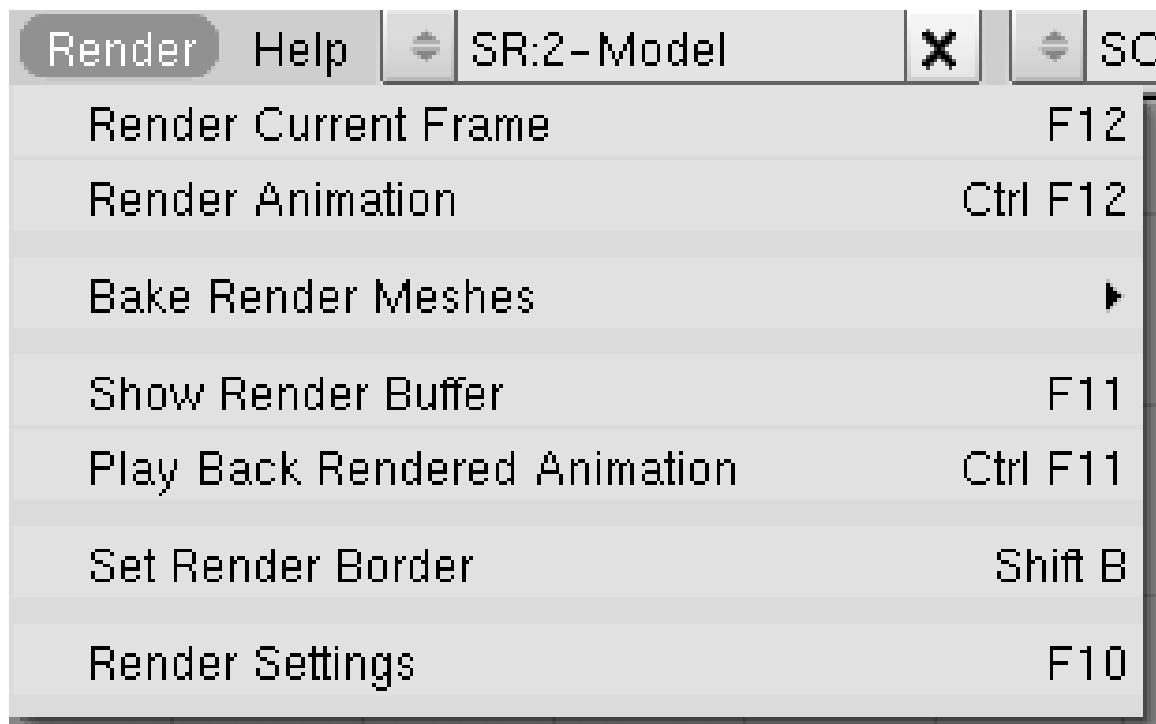


Figure 27

² Chapter 10 on page 49

You can move through items in a menu by:

- moving the mouse pointer up and down

or

- pressing  and 

You can enter a submenu either by:

- moving the mouse pointer to the right

or

- pressing 


You can leave a submenu either by:

- moving the mouse pointer to the left

or

- pressing 

To initiate a menu action, you can either:

- click 


or

- press 

You can escape from a menu either by:

- moving the mouse pointer away from the menu

or

- pressing 

For each menu, Blender remembers which item you chose last time and highlights it for you the next time you enter the menu.

9.3.1. Notation

Menu notation is fairly self-explanatory.

 +  *Mesh* → *UV Sphere*

Means:

1. Press Shift+A
2. In the menu that pops up, move through the items until *Mesh* is highlighted
3. Enter the *Mesh* submenu
4. Move through the items until *UV Sphere* is highlighted
5. Press Enter or click the left mouse button to initiate the action

9.4. Additional Resources

- The Blender Manual page on keyboard and mouse at <http://wiki.blender.org/index.php/Doc:Manual/keyboardandmouse>

--Nkansahrexford³ (discuss⁴ • contribs⁵) 01:37, 11 February 2012 (UTC)Nkansahrexford

³ <http://en.wikibooks.org/wiki/User%3ANKansahrexford>

⁴ <http://en.wikibooks.org/wiki/User%20talk%3ANKansahrexford>

⁵ <http://en.wikibooks.org/wiki/Special%3AContributions%2FNkansahrexford>

10. Non-standard equipment

This module is applicable only to users with non-standard input devices. If you have both:

- a three-button mouse

and

- a keyboard with a numpad,

then you can skip this module.

10.0.1. Keyboards lacking a numpad

Most modern laptops have a **pseudo-numpad**, a set of keys in the main keypad which double as a numpad. The keys typically used for this purpose are:

7KEY	8KEY	9KEY	0KEY
U	I	O	P
J	K	L	;
M	.KEY	.KEY	SLASH

When used as a pseudo-numpad, these keys typically act as the following keys from a true numpad:

NUM7	NUM8	NUM9	NUM/
NUM4	NUM5	NUM6	NUM*
NUM1	NUM2	NUM3	NUM-
NUM0	NUMENTER	NUM.	NUM+

The numpad functions of these keys can often be toggled with **F11** or **NUMLOCK** on PCs or with **F6** on Macs. Alternatively, you can often temporarily activate the numpad behavior by holding down **FN**.

If your keyboard has the alternate labellings but you don't know how they work, consult your laptop's owner manual.

As a last resort, you can use the "Emulate Numpad" feature of Blender. This will allow you to use the normal numeric keys as if they were numpad numerics. Instructions for enabling this feature may be found in the "User Preferences Windows" module .

Blender uses the numeric keypad quite a bit. If you envision using your laptop for this kind of work, it may be worth investing in a *USB Numeric Keypad*. On eBay, prices for simple external numpads start around \$10 USD.

10.0.2. Non three-button mice

For single-button mouse users, make sure that *View & Controls (Input for Blender 2.54)* (under "User Preferences" on the left-most drop-down menu) → *Emulate 3 Button Mouse* is enabled.

On many computers with two-button mice, **MMB** can be emulated by simultaneously clicking **LMB** and **RMB**. On Windows machines you'll need to set this up in the mouse settings in your Control Panel. On a Mac, open the *Keyboard and Mouse* preference pane and enable *Use two fingers to scroll*. Alternatively, by selecting *Emulate 3 Button Mouse* under *User Preferences*, **MMB** can be emulated by simultaneously clicking **ALT** and **LMB**.

Recent IBM Thinkpad laptops allow you to disable the 'UltraNav' features of the middle mouse button in order to use it as a 'normal' third button. Alternatively, some laptops allow areas (called gestures) on the movement pad to act as **MMB** or **RMB**, and these can be set up in the Control Panel in the Mouse Pointer options, selecting gestures and editing features there.

Apple single-button mouse

Apple single-button mouse substitutions Notation	Single-button Substitute	Description
LMB	MB	the Mouse Button
RMB	CMD + MB	Apple key + the Mouse Button
MMB	ALT + MB	Option (Alt) key + the Mouse Button

While Mac OS X natively uses both the **CTRL** + **MB** and **CMD** + **MB** to emulate **RMB**, recent Blender releases for Mac OS X use only **CMD** + **MB** for this purpose. This behavior is documented in the *OSX Tips* file that comes with the Mac version. You can also set in System Preferences for the mouse to sense a right-click.

Note also that in the new, "unibody" design, the mouse button is under the trackpad, and the shortcut for **RMB** is clicking with two fingers simultaneously, which can be enabled in the System Preferences.

10.0.3. Tablet PCs

To get the effect of **MMB** in a viewport, drag your pen around while holding down the **ALT** key.

10.0.4. Additional Resources

- keyboard and mouse¹ — Blender Manual²

¹ http://wiki.blender.org/index.php/Doc:Manual/Interface/Keyboard_and_Mouse

² <http://wiki.blender.org/index.php/Doc:Manual>

11. Operating System specific notes

This tutorial covers user-interface issues that are specific to particular operating systems or window managers. Read the section that applies to your computer; you may skip the rest.

11.1. GNU+Linux

ALT + **LMB** is used for changing the angular view on two angular axes of the 3D View window, if **ALT** + **LMB** moves the current window, then there's a conflict with your window manager. You can resolve the conflict or use **CTRL** + **ALT** + **LMB** or **MMB** instead (Also, you may have activated Compiz->Rotate Cube. Default configuration for rotating the Cube is also **CTRL** + **ALT** + **LMB**; you may have to change this binding to an alternative configuration.). If you are running KDE this can be resolved by: **RMB** on the title bar of the main Blender window → select *Configure Window Behavior* → go to *Actions* → *Window Actions* → in the *Inner Window, Titlebar and Frame* section → select the Modifier key to be **ALT** and set all the select boxes beneath it to *Nothing*. An alternate method within KDE might be to **RMB** click on the title bar of the main Blender window; then select *Advanced* → *Special Application Settings...* → *Workarounds* and then click *Block global shortcuts* with *Force* selected and checked.

In Gnome, Click *System* → *Preferences* → *Window Preferences*. Look for the last three options *Control*, *Alt* and *Super*. Select *Super*. Now you can press and hold **CMD** to drag windows around, and use **CTRL** and **ALT** as normal.

11.1.1. KDE

Under KDE, **CTRL** + **F1** through **CTRL** + **F4** are by default configured to switch to the corresponding one of the first four desktops. You can change this in System Settings.

11.1.2. Gnome

You'll want to disable the *Find Pointer* functionality in Gnome, which will impair your ability to use certain functions such as *Snap to grid* and the lasso tool. If your mouse pointer is being highlighted when you press and release **CTRL**, go to: *Mouse* in Gnome's *Desktop Settings* and uncheck the box *Find Pointer*.

11.1.3. Ubuntu

As of Ubuntu versions prior to about 09.10 (“Karmic Koala”), there was a known incompatibility between Blender and the Compiz Fusion accelerated (OpenGL) window manager used in Ubuntu.

By default, Compiz Fusion is enabled in Ubuntu, causing the problems to manifest themselves in Blender as flickering windows, completely disappearing windows, inconsistent window refreshes, and/or an inability to start Blender in windowed mode.

The fix for this is simple. Install compiz-switch (might be in universe). Go to *Applications* → *Accessories* → *Compiz-Switch*. This will disable compiz temporarily. Do the same to turn compiz back on when you're done using Blender.

This is no longer needed for current releases of Ubuntu.

11.2. Mac OS X

On Macs with the new thin Apple Keyboard, you may need to press **FN** in order to use the **F1** through **F4** and **F7** through **F12** keys. To expand a section in Blender, you should usually press **CTRL** + **UPARROW**, on a Mac if "Spaces" is enabled you may have to use **CTRL** + **ALT** + **UPARROW**.

11.3. Microsoft Windows

11.3.1. Two Ways to Launch Blender

Blender requires a console for displaying error messages, so if you launch Blender by means of an icon, two windows will appear: the graphical user interface plus a console window. Closing either window will terminate Blender. These windows are indistinguishable in the Windows taskbar in versions of Windows before Windows 7¹, which leads to confusion. Also, launching this way does not provide any way to pass command-line arguments to Blender.

Launching Blender from a command prompt is extra work, but it overcomes these issues:

1. *Start* → *Run...*
2. enter `cmd`
3. enter `cd c:\Program Files\Blender Foundation\Blender`
4. enter `blender`





EDIT: Blender 2.6 doesn't have this problem, and hides the console window by default. You can show it by clicking *Help* > *Toggle system console*

11.3.2. Sticky Keys





Pressing **SHIFT** five times in a row may activate **StickyKeys**, an accessibility option which alters how the computer recognizes commands. If a StickyKeys dialog box appears, you should **LMB** the "Cancel" button.



If you don't need the accessibility features, you can disable sticky keys:

¹ <http://en.wikipedia.org/wiki/Windows%207>






1. *Start* → *Control Panel*
2. double-click on *Accessibility Options*
3.  the *Keyboard* tab
4. for each of the options *StickyKeys*, *FilterKeys*, and *ToggleKeys*:
 - a) clear the *Use ...* checkbox
 - b)  the *Settings* button
 - c) uncheck the *Use Shortcut* checkbox in the settings
 - d)  the *OK* button for the settings
5.  the *OK* button for *Accessibility Options*

11.3.3. Multiple Keyboard Layouts

On systems with multiple keyboard layouts, pressing  +  can alter the layout. (For instance, it might change from QWERTY to AZERTY or vice versa.) Because of this issue, *Noob to Pro* avoids  +  hotkeys.

If you find your keyboard layout altered, press  +  again to change it back.

You can also disable the hotkey:

1. *Start* → *Control Panel*
2. double-click on *Regional and Language Options*
3.  the *Languages* tab
4.  the *Details* button
5.  the *Key Settings* button
6.  the *Change Key Sequence* button
7. uncheck the *Switch Keyboard Layout* checkbox
8.  the *OK* button

11.3.4. Additional Resources

- Input method editor keyboard shortcut (CTRL+SHIFT+0) switches the input language in Vista² — Microsoft Support Knowledge-Base
- w:StickyKeys³

² <http://support.microsoft.com/kb/967893>

³ <http://en.wikipedia.org/wiki/StickyKeys>

12. Blender Interface

i Information

Applicable Blender version: 2.57

Here's a screenshot to give you a preview of Blender's interface.

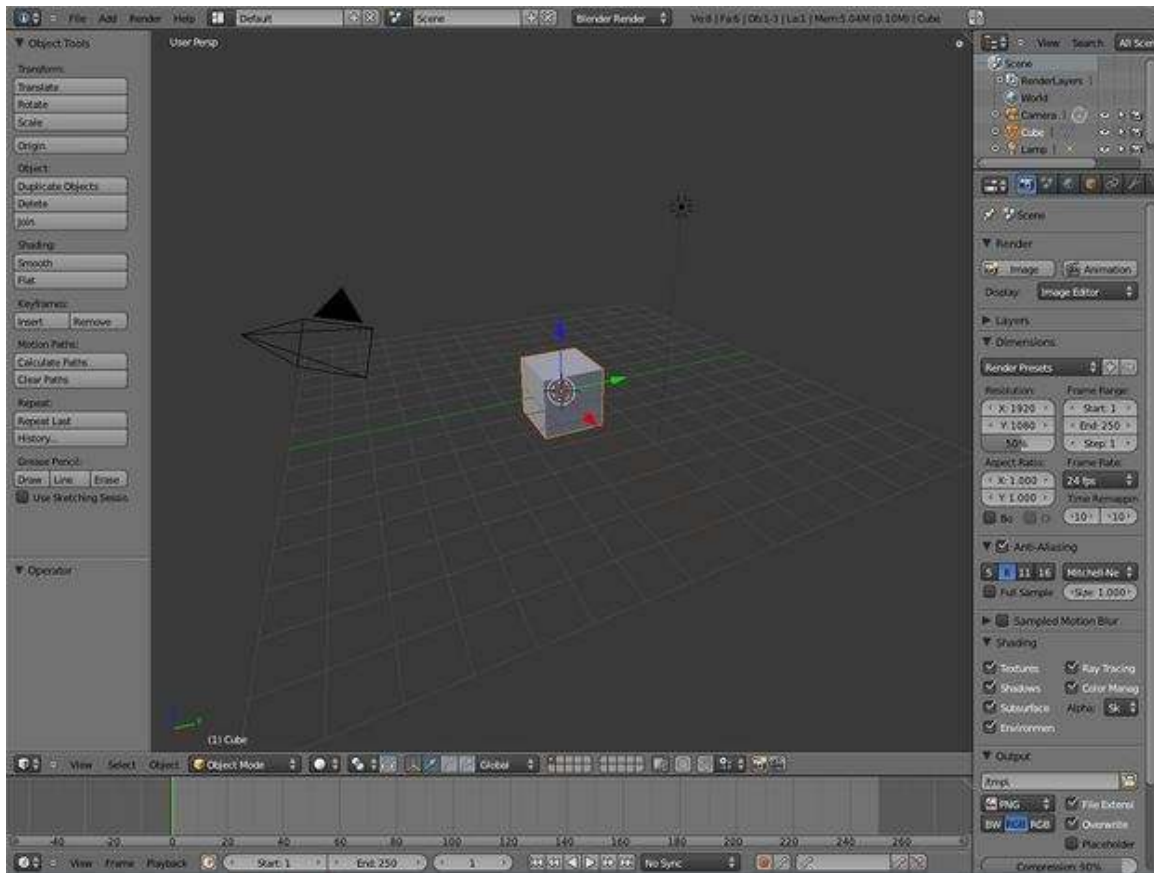


Figure 28

13. Blender Windowing System

i Information

Applicable Blender version: 2.59

The Blender user interface may appear daunting at first, but don't despair. This book explores the interface one step at a time.

In this module, you'll learn about Blender windows:

- recognizing windows and their headers,
- the different types of windows,
- how to activate and resize windows,
- how to split and join windows.

You'll also practice launching and leaving Blender.

13.1. An Interface Divided

Blender's user interface is divided into rectangular areas with rounded corners. These areas are called **windows** (or sometimes, **areas** or **Workspaces**).

If you haven't already launched Blender, go ahead and do so. You should soon see something that resembles the following.

Blender has had some major changes to its user interface (UI) since version 2.4x. Some of these changes include moving buttons and changing the space bar hot key from the “add menu” to the “search menu” (Shift+A is now the “add menu” hot key). This is important to know when trying to follow tutorials.

Other changes include the addition of the tool bar and window splitting widget. The tool bar widget (indicated by a plus sign) opens hidden tool bars. The object tool bar can be toggled on and off by pressing the T key. The 3d window tool bar can be toggled on and off by pressing the N key. The splitting window widget allows you to split and join windows. Blender 2.59 is shown below.

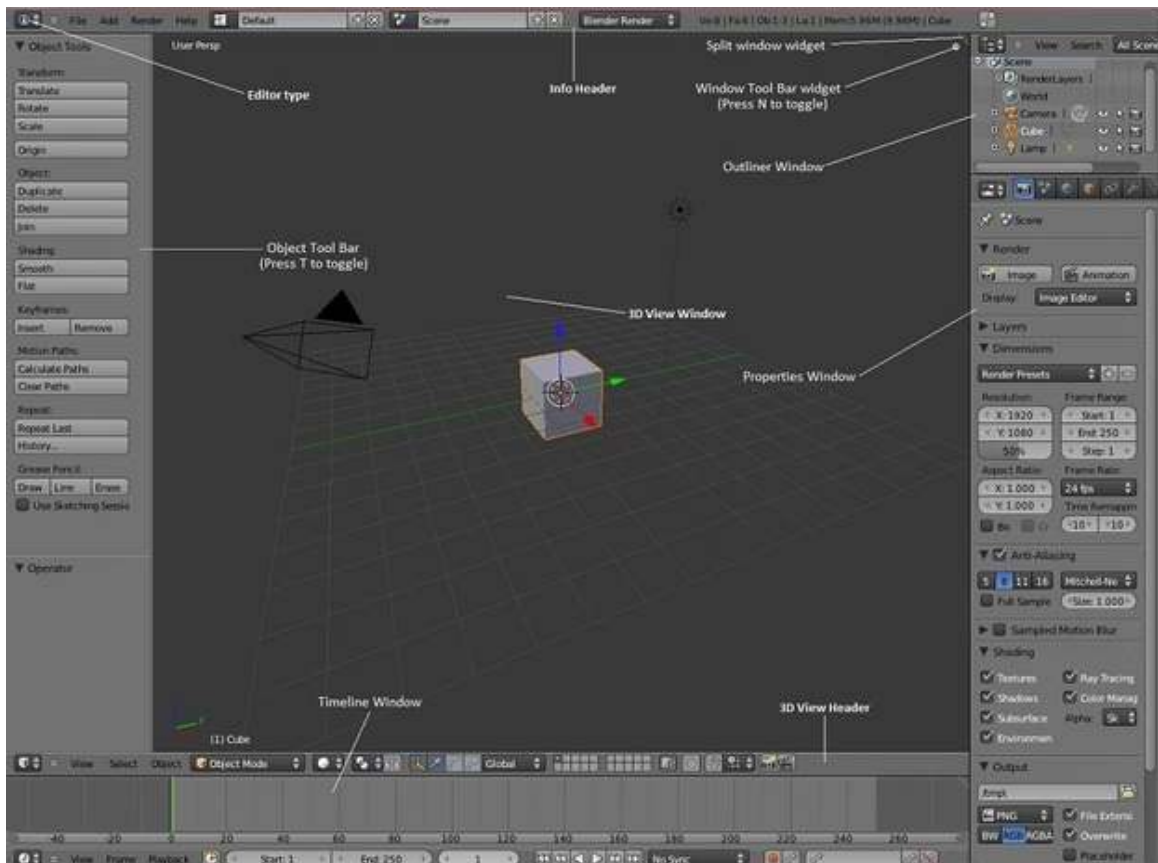


Figure 29 Blender 2.5x

- You may be running a different version of Blender - maybe a higher version. The screenshot was made using the 2.59 release.

If you're running an older version, you should probably upgrade. Download instructions are in the Introduction¹.

- The user-interface settings on your computer may have been changed. Try resetting the user interface with *File* → *Load Factory Settings*.
- If you're running a stable (not "beta") release of Blender later than 2.57, *please* consider uploading a new screenshot and updating the wikibook!

To take a screenshot in Blender, press **[ALT] + [F3]**, and click *Make Screenshot*. This will record what's on your screen until you click the red *Close* button on the info header. The screenshots will be saved in the *tmp* folder.

13.2. Window Headers

Did you find all five headers?

¹ Chapter 1 on page 5

Every Blender window has a header. A header can appear at the top of the window, at the bottom of the window, or it can be hidden. Let's take a closer look at the headers.

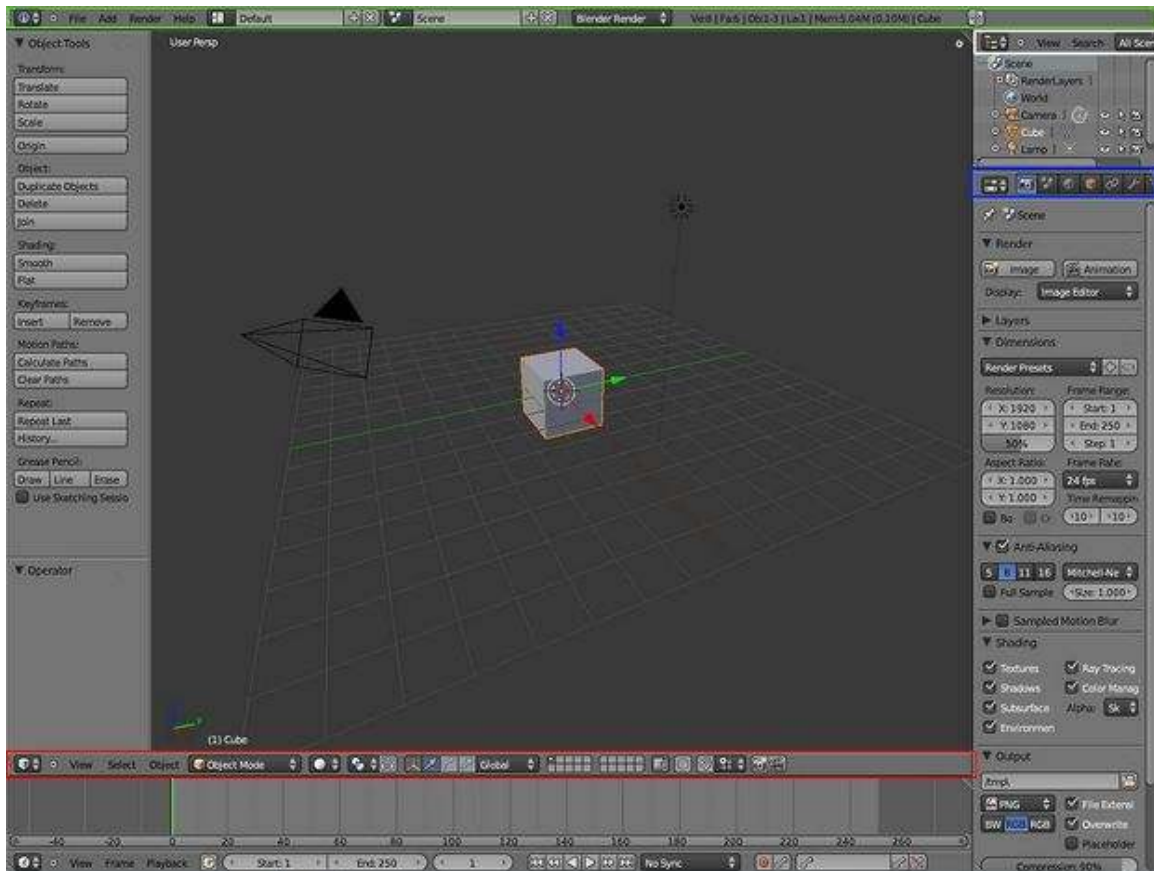


Figure 30 The header of the Info window is outlined in green .

The header of the 3D View window is outlined in red . Note that it runs along the *bottom* of the 3D View window, not the top.

The header of the Properties window is outlined in blue .

The header of the Outliner window is outlined in white.

The header of the Timeline window is the one on the bottom (not outlined)

13.3. Window Types

Blender has 15 types of windows. You've just encountered the Info, 3D View, Properties, and Outliner windows. The rest will be introduced as needed in later modules.

Every window header in Blender has an icon at the left end to indicate the window type. For instance:



Figure 31

= Info

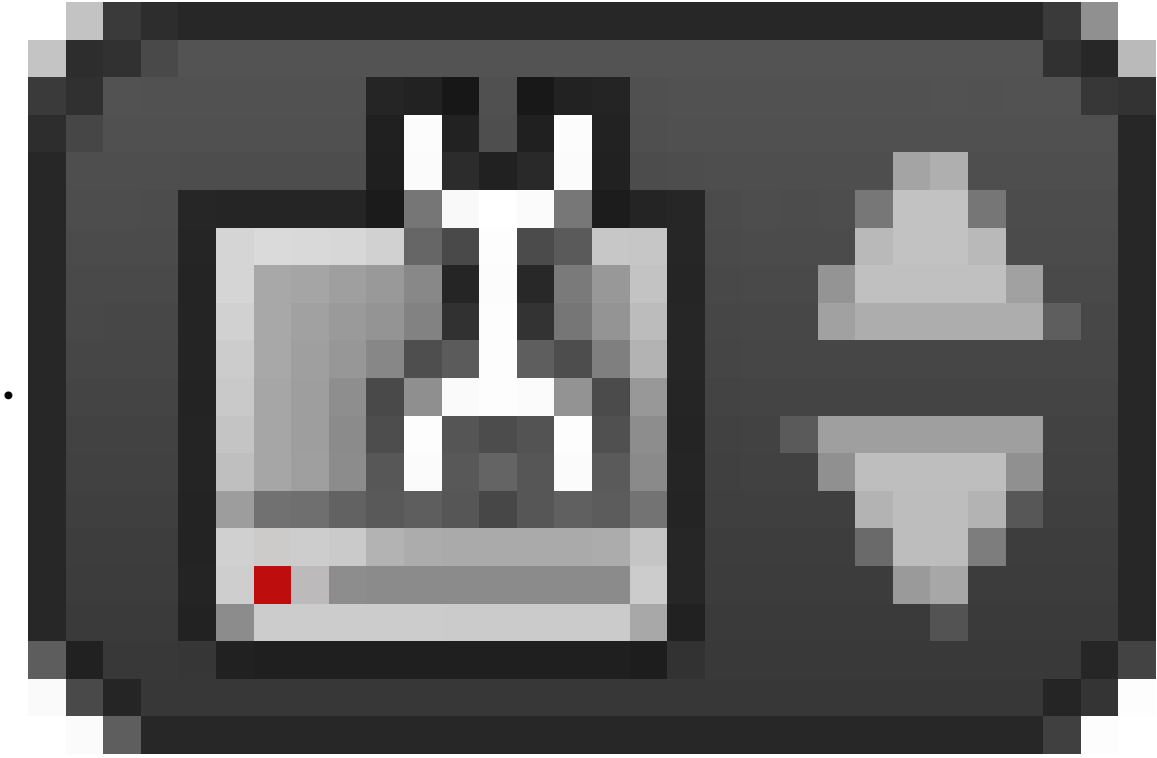


Figure 32

= User Preferences

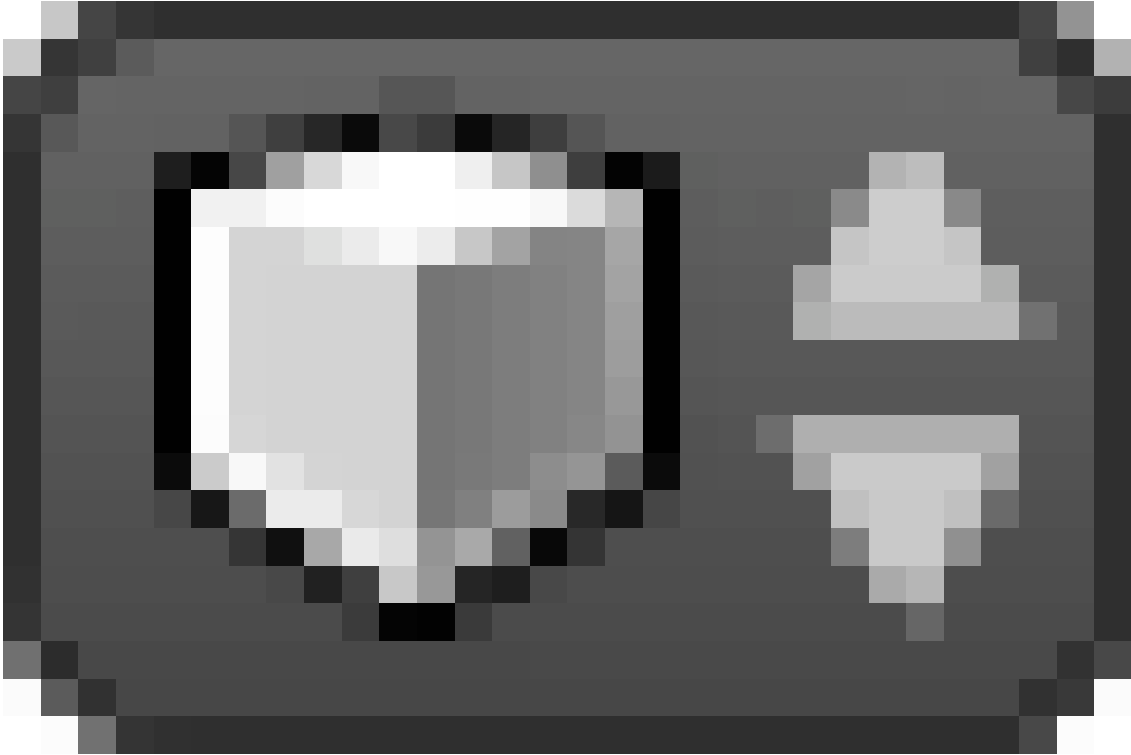


Figure 33

= 3D View



Figure 34

= Outliner

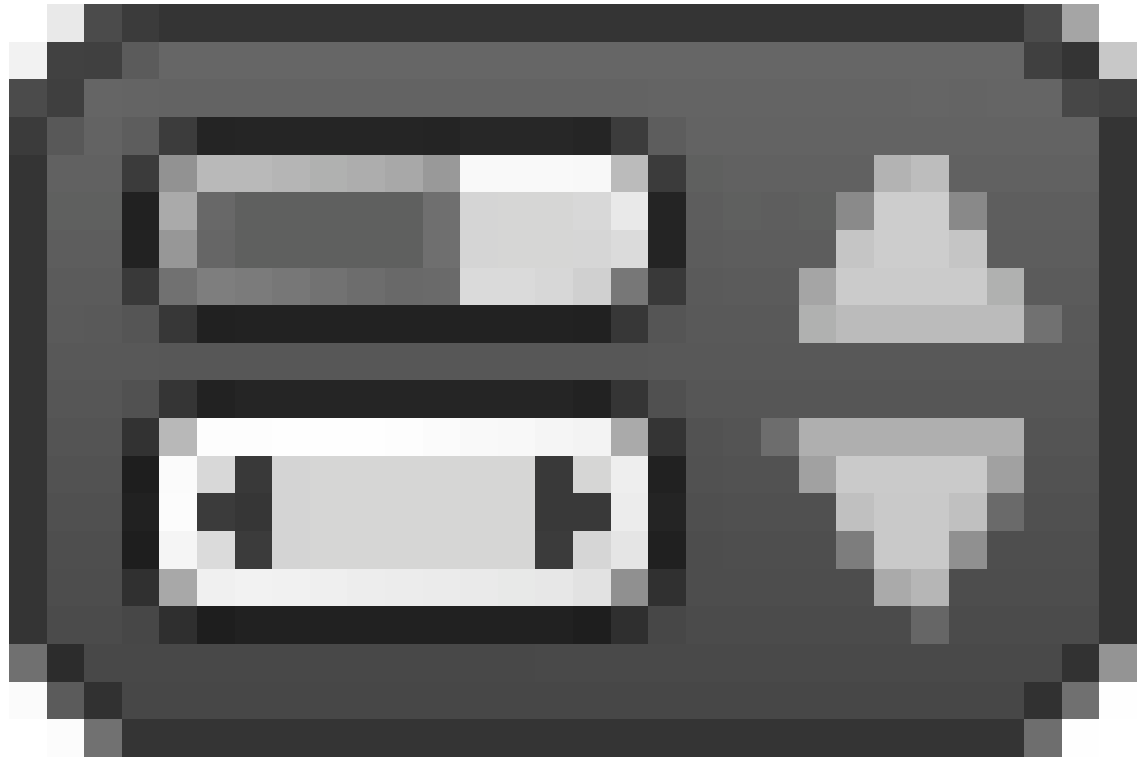


Figure 35

= Properties

If you **LMB** on the icon, a menu will pop up. (If you don't know what **LMB** means, please review the Keystrokes, Buttons, and Menus module².)

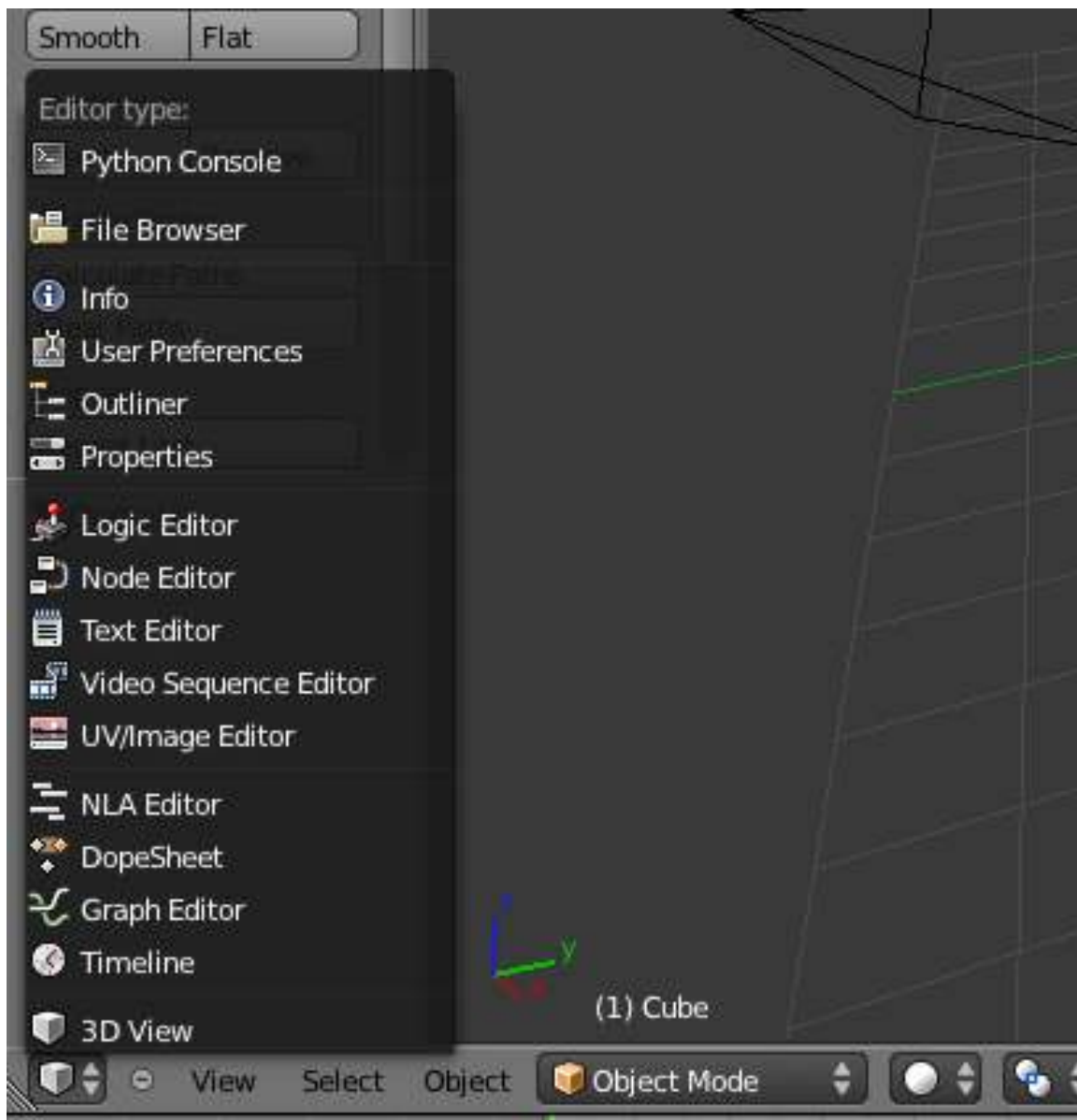


Figure 36 By matching the icon in the header to the icons in the menu, you can tell that the window here is a 3D View window. The menu can be used to alter a window's type. In this screenshot, the user is about to change the window into a Properties window.

Note:

Any window can be changed to any type.

Note:

Blender doesn't mind if there are multiple windows of the same type.

If you've changed any window's type, please change it back (or reload the factory settings with *File* → *Load Factory Settings*) before continuing with this tutorial.

13.4. The Active Window

The **active window** is the one that will respond if you press a key. Exactly one Blender window is active at any given time.

The active window is usually the one containing the mouse pointer. (Blender uses a "focus follows mouse" user interface model. When a hotkey fails to work as expected, it is often because the mouse pointer has strayed into a neighboring window.) To change the active window, simply move the mouse pointer into the window you wish to activate.

Practice changing the active window by moving your mouse between the 3D View and the Timeline windows. The Timeline window is exactly below the 3D View header. At this point, it is worth mentioning that the header for the 3D View window and Timeline window is at the **BOTTOM** of its own window instead at the top as the name "header" implies.

Note:

When a window becomes active, its header gets brighter.

13.5. Resizing Windows

Resizing windows is easy.

13.5.1. Dragging on a Border



Figure 37 Press and hold **LMB**.
Drag with the mouse to move the border up and down.
When the border is where you want it, release **LMB**.

Whenever you increase the size of one window, you decrease the size of another. That's because Blender has a **non-overlapping window interface**: unlike many other programs, it does not permit windows to overlap.

13.5.2. Maximizing a Window

Another way to resize a window is to maximize it. When Blender **maximizes** a window, it makes it as big as possible. The previous window configuration is saved.

- To maximize the active window, press **CTRL** + **UPARROW** , **CTRL** + **DOWNARROW** or **SHIFT** + **SPACE** .
- When a window is maximized, use **CTRL** + **UPARROW** , **CTRL** + **DOWNARROW** or **SHIFT** + **SPACE** to restore the previous (unmaximized) window configuration.

Practice maximizing and un-maximizing the 3D View and Timeline windows.

Note:

If you are running a version of Blender before 2.57, you cannot maximize a User Preferences window.

13.6. Splitting a Window

Update (version 2.57): Now the process of splitting/Joining does not work through a menu. Tab is also not needed. Splitting is done by pulling from the edge of the window towards the interior of the same (towards a second window of a different type, causes the first window to override the second). Joining is done by pulling towards a second window of the same type (a big arrow appears). The splitting direction is defined by the direction in which you move your mouse.

Information

Applicable Blender version: 2.49

Splitting a window is just as easy as resizing. A split always results in two windows of the same type.

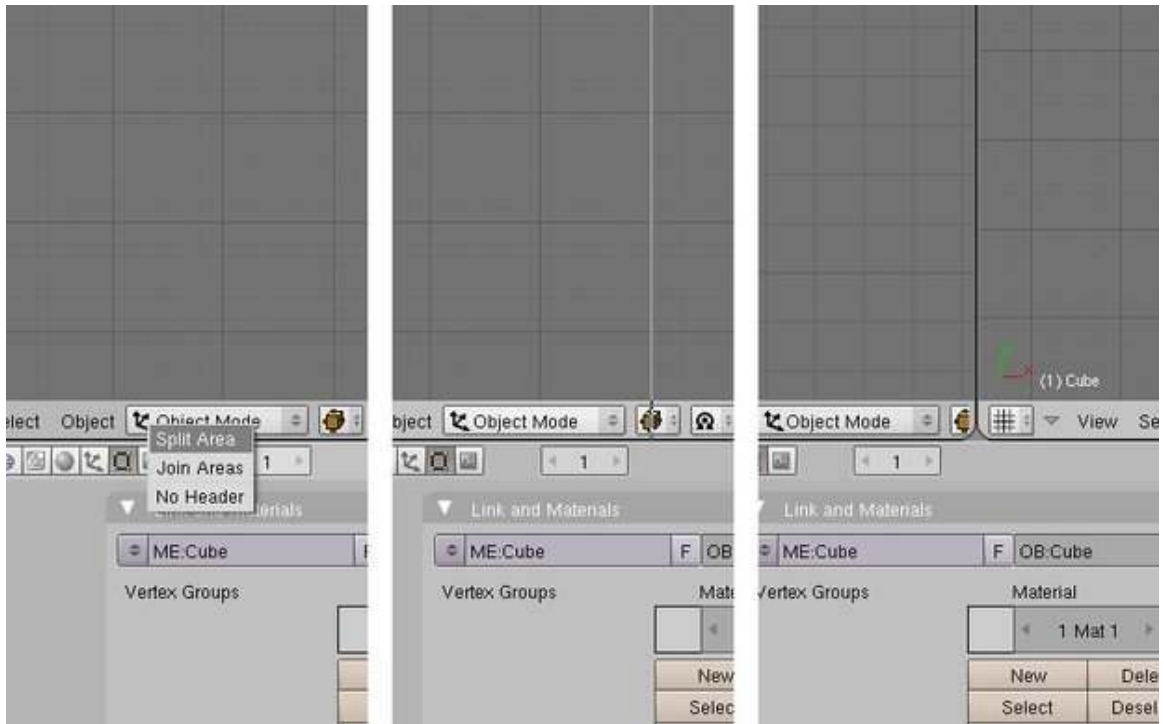


Figure 38 A menu will pop up. Select *Split Area* from the menu using **[LMB]**.

(If you don't know how to produce **[RMB]** on your computer, please review the "Non-standard Input Devices"³ module.)

Splitting on a horizontal division, as we have done, will give you two windows side-by-side horizontally. Splitting a window on a vertical division will give you two windows stacked vertically.

To switch between vertical and horizontal splitting during Step 3, press **[TAB]**.

13.7. Joining Windows

Combining adjacent windows is as easy as splitting them. We'll rejoin the window we just split.

³ Chapter 10.0.2 on page 50

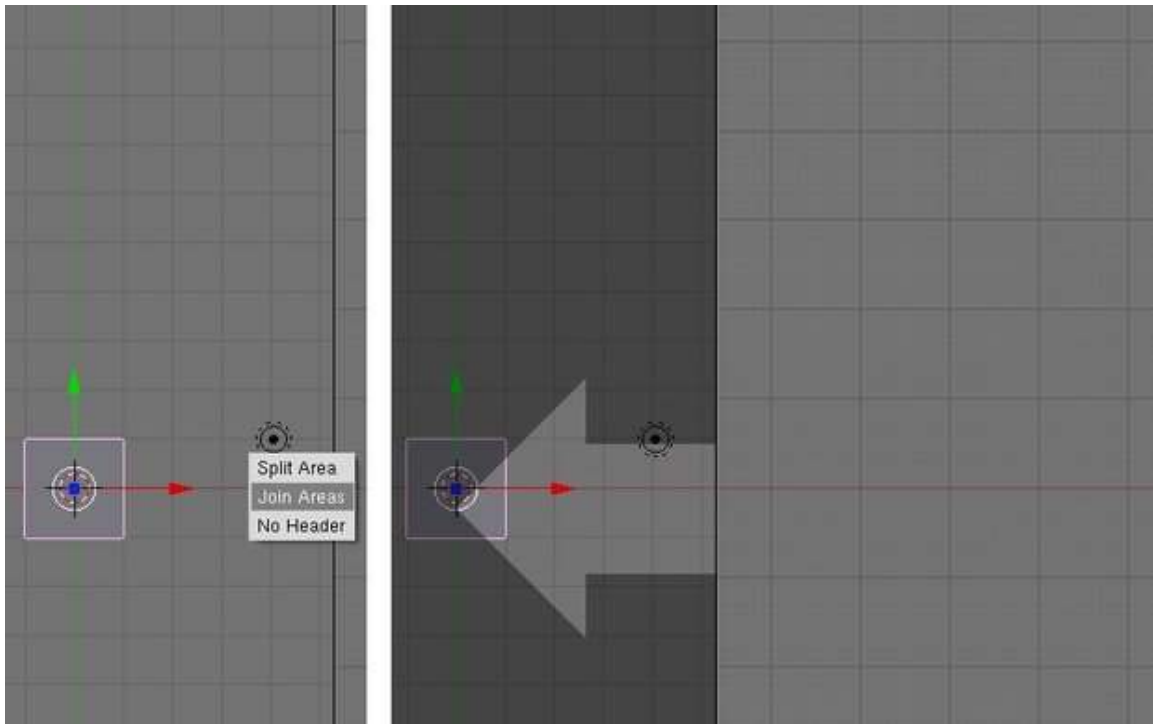


Figure 39 A large grey arrow will appear to indicate the merge direction. By moving the mouse pointer between the two windows, you can change the merge direction.

Joining to the left will destroy the window on the left, while joining to the right will destroy the window on the right. This becomes important when joining different window types.

Note:

If you **[RMB]** on a border and Blender doesn't offer you the *Join Areas* option, it's because the border touches more than two windows. You'll have to select the window you want removed using a different border.

13.8. Leaving Blender

To close down Blender:

1. If there's a tool active, press **[ESC]** to exit the tool.
2. Press **[CTRL] + [Q]**. This brings up an *OK?* menu.
3. Confirm *Quit Blender* by clicking **[LMB]** or pressing **[ENTER]**.

Note:

Blender will not prompt you to save your work. However, you can easily restart Blender and then pick up where you left off using *File* → *Recover Last Session*.

13.9. Additional Resources

- **YouTube video on Splitting and Joining Windows in 2.49** at www.youtube.com/watch?v=uYb1j8X-{}ulc
- **YouTube video on Splitting and Joining Windows in 2.59** at www.youtube.com/watch?v=mGK1gwFhx9M
- **The Blender Manual page on window types** at <http://wiki.blender.org/index.php/Doc:Manual/windowtypes>
- **The Blender Manual page on changing window frames** at <http://wiki.blender.org/index.php/Doc:Manual/changingwindowframes>

14. User Preferences Windows

i Information

Applicable Blender version: 2.57

In this module, we'll take a closer look at the User Preferences window. In the process, you'll encounter three different user-interface controls: radio buttons, toggle buttons, and sliders.

14.1. Saving User Preferences

Most other applications have a place to keep user-configured settings (including document defaults), separate from any actual documents the user creates. Blender works in a slightly different fashion: all your user-configured settings are saved in *every* document you create. Each time you create a new Blender document, it actually just reloads your *default* document, which is called .B.blend. To save your current state as the default document, press **CTRL** + **U**. This will save *everything* you've done to the current in-memory document, including objects and materials created, to .B.blend.

14.2. Accessing the User Preferences

First we must open the User Preferences window. There are 3 ways to do this:

- Click **LMB** *File* → *User Preferences...*
- Change the window type of the top header to *User Preferences* and drag the header down.
- Press **CTRL** + **ALT** + **U**, which will open the User Preferences into a separate window which you can resize at will.

The User Preferences window should look something like the screenshot below.



Figure 40

14.3. Configuring Your Preferences

So as to get to modeling and rendering sooner, this tutorial will cover only a few of the many preferences which you can set.

14.3.1. Auto Save

Auto Save is a very useful feature that automatically saves the current .blend after a specified period of time. The settings are:

- *Auto Save Temporary Files:* This enables/disables the auto save feature.
- *Timer (mins) slider:* This specifies the time in minutes between each auto save.

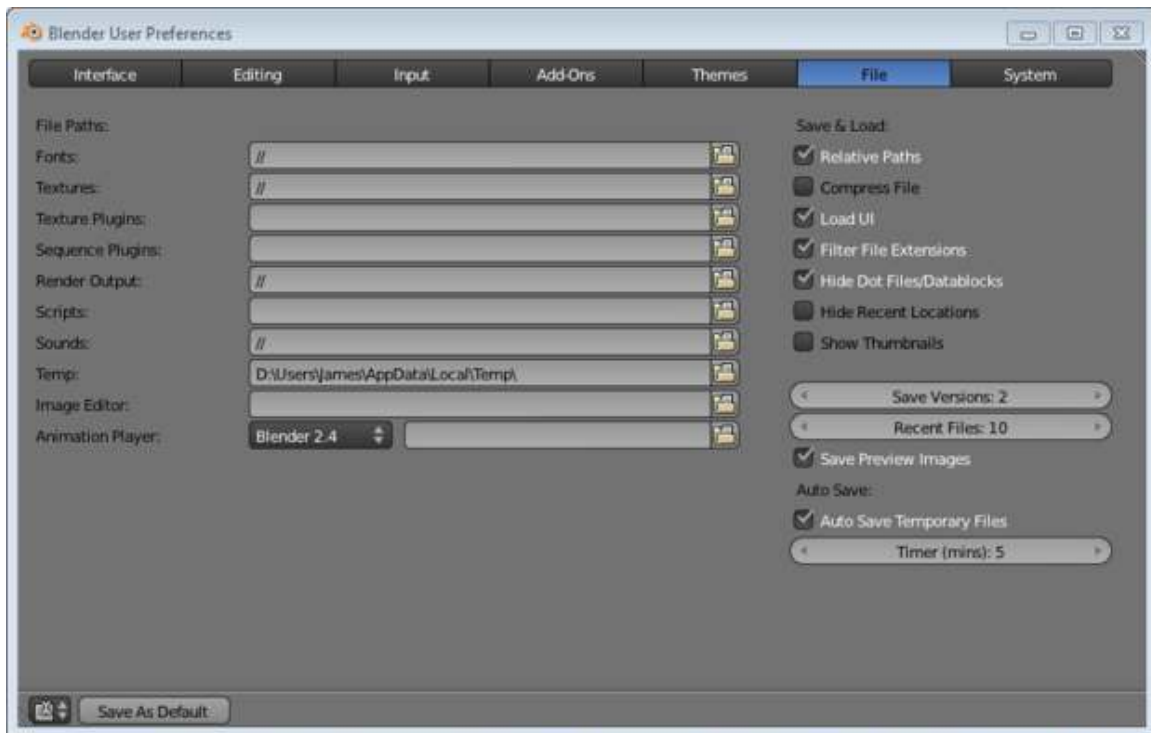


Figure 41

14.3.2. Number of Undo Levels

Next we'll look at the undo settings. By default, Blender remembers your last 32 actions and allows you to undo them one at a time by pressing **CTRL** + **Z**. If your computer has plenty of memory, you may wish to increase that number. If it has relatively little memory, you might consider decreasing it to 10 or 20. The *Memory Limit* slider specifies the amount of RAM(in megabytes) to use for storing the undo levels. Undo level "0" is unlimited.

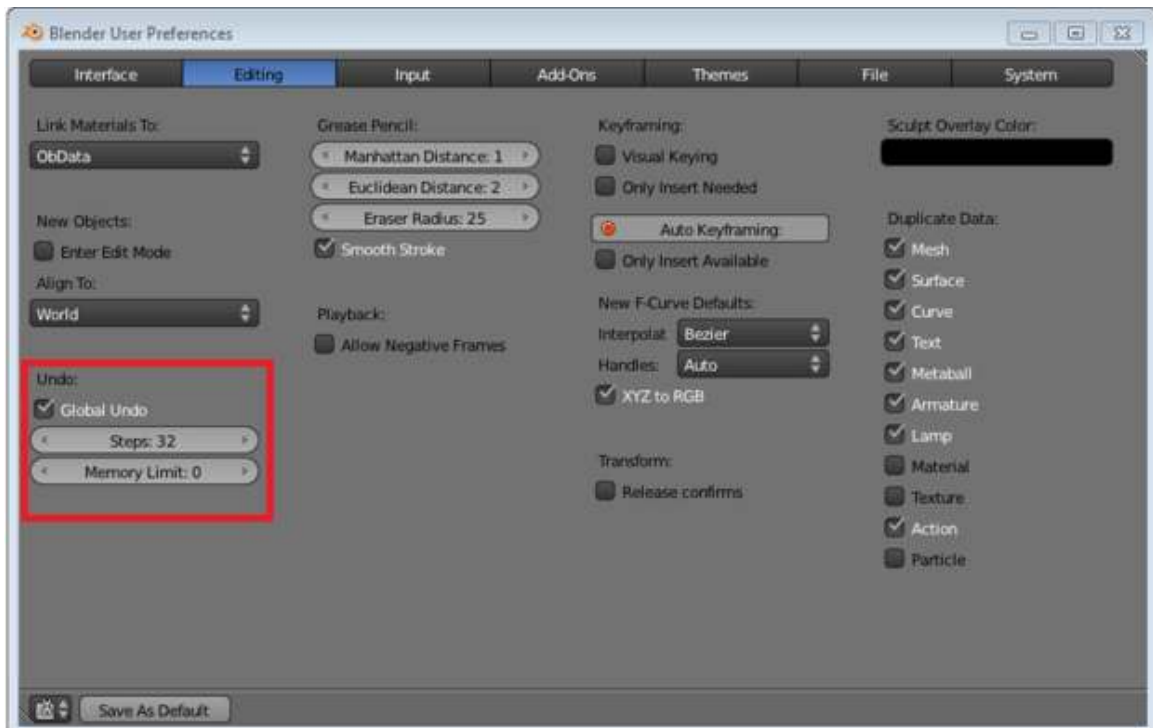


Figure 42

14.3.3. Numpad Emulation

Blender uses numberpad keys (such as **NUM7**) to control the 3D View and ordinary numeral keys (such as **7KEY**) to change layers. If you are working on a laptop or if you find the numberpad inconvenient, you can select *Emulate Numpad* to reassign the 3D View controls to the ordinary numeral keys.

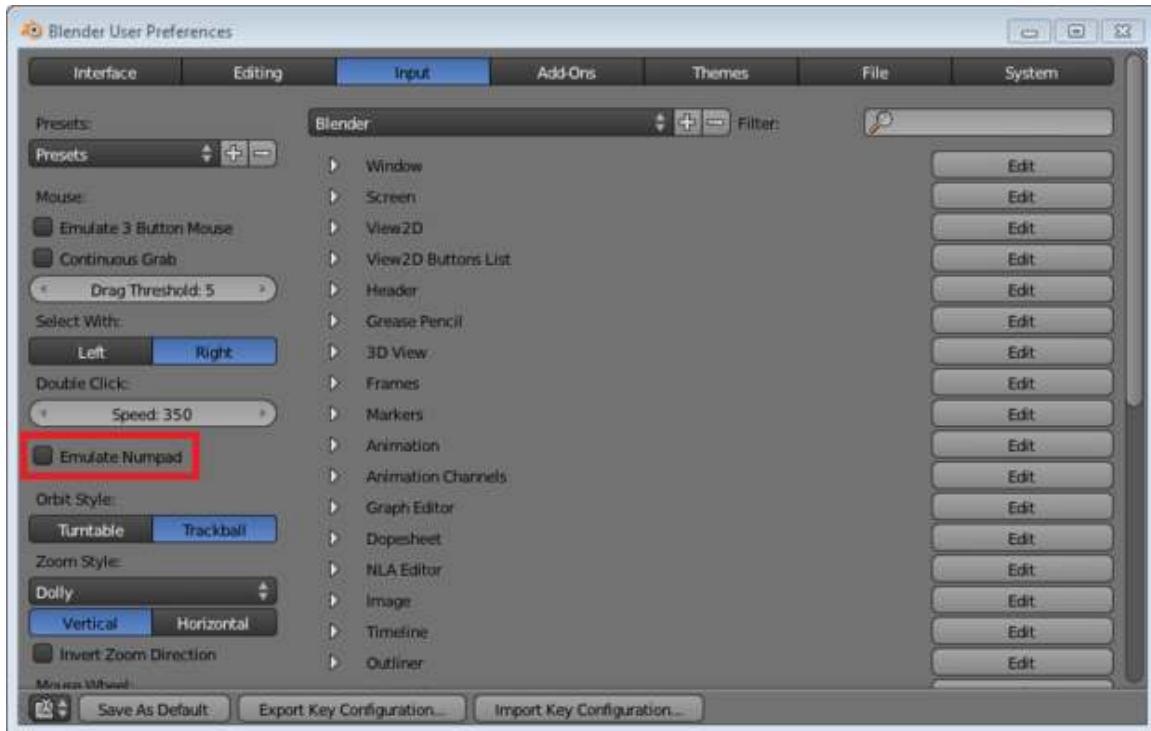


Figure 43

If you ever need to restore Blender to its factory settings, you can either delete your personal ".blend" file and then restart Blender or click **[LMB]** *File* → *Load Factory Settings*

Note:

The second method only affects the current session. To make the settings persist, you would need to save them with **[CTRL] + [U]** or *File* → *Save User Settings*.

14.4. Additional Resources

- The Blender Manual page on Configuration at <http://wiki.blender.org/index.php/Doc:Manual/Configuration>
- The tutorial on Non-standard Equipment¹ describes other workarounds for numpad issues.

¹ Chapter 10 on page 49

15. Buttons Windows

i Information

Applicable Blender version: 2.57

The Properties window is where you will find most of the functions that Blender can perform with objects and materials, animation, rendering and so on. Here is also the area where you will see the greatest changes from earlier versions of Blender (where it was called the Buttons window). Hopefully you'll agree that the new layout makes it much easier and quicker to find things!

In the header of the Properties window, you will see a row of buttons that looks like this:



Figure 44

The actual icons will vary depending on the type of object selected in the 3D view; also in the default layout, the Properties window may be too narrow to show the entire row, in which case you can either widen the window, or click drag across the buttons with **[MMB]** to scroll the button row. Each of these buttons gives you access to a different *context*, or subsection of the Properties settings. Unlike older Blenders, there are no more “subcontexts”—no additional buttons will appear in the header when you click any of these. At times, the row may appear not shown unless you increase the width of that panel. Well, Blender has thought about that and solved that already. You can either click on your **[LMB]** and drag the row sideways depending what you want to let appear, either to the right or left. Or, the other way out is to hover your mouse over the row, and scroll your **[MMB]** to bring the hidden icons to view. Okay, let's move on!

16. The Contexts

16.1. Render Context File:Blender255RenderContextButton.png

In here are the settings that control rendering of the final images: which layers to render, what resolution to use, output format, performance, post processing, etc.

16.2. Scene Context File:Blender255SceneContextButton.png

In here are some settings for use when previewing the scene on-screen, and choosing which camera to use for rendering.

16.3. World Context File:Blender255WorldContextButton.png

In here are settings that govern the surroundings in which the model is rendered: background sky color, mist and star settings, environment lighting etc.

16.4. Object Context File:Blender255ObjectContextButton.png

In here are settings that apply to all kinds of objects: overall transformations, layer assignments, grouping etc. The settings shown here (and any changes made) apply to the last object selected. This is also the case for the following object-specific contexts.

16.5. Object Constraints Context

File:Blender255ObjectConstraintsContextButton.png

In here are settings that limit the motion of the object for animation purposes. The limits can also be tied to the motion of other objects in various ways.

16.6. Object Modifiers Context

File:Blender255ObjectModifiersContextButton.png

In here are settings for applying modifiers to the object geometry. Note that lamps, cameras and empty objects cannot have modifiers.

16.7. Object Data Context

File:Blender255ObjectDataContextButtons.png

In here you will find settings specific to the type of object: mesh vertex groupings, text font, lamp settings, camera settings, all that kind of things. This is reflected in the icon, which changes according to the type of object selected.

16.8. Material Context File:Blender255MaterialContextButton.png

The material settings for an object control how it looks: its colour, whether it has a shiny or dull surface, how transparent it is, and so on.

16.9. Texture Context File:Blender255TextureContextButton.png

The texture settings specify patterns that break up the uniform appearance of a material: these patterns can affect the colour of the material, or give it a rough surface, or modify it in other ways.

16.10. Particles Context File:Blender255ParticlesContextButton.png

An object can be set to emit particles, like smoke, flames or sparks. But the concept of “particles” (and the underlying algorithms) also includes the generation of hair or fur.

16.11. Physics Context File:Blender255PhysicsContextButton.png

In here you will find settings that control how the object reacts to forces like objects in the real world: whether it behaves like a rigid body that keeps its shape but can be knocked around, cloth, something soft e.g a pillow, or a flowing liquid.

17. Where Did The Old Stuff Go?

For those used to older versions of Blender, the old Logic Context has been moved into its own window type, the Logic Editor.

The old Script Context has gone: Python scripting is now much more closely integrated into the Blender UI, and old scripts will not work anyway.

The functions of the other contexts have been rearranged into the new contexts as listed above. Once you get used to the new arrangement, it should make much more sense than the old one.

18. 3D View Windows

Information

Applicable Blender version: 2.57

3D View windows in Blender are used to visualize 3D scenes. You'll do a lot of work in these windows, so you'd better learn your way around.

In this module, you'll learn:

- to recognize 8 things commonly seen in viewports
- to tell which mode Blender is in
- how to change viewport options and viewpoints
- how to undo changes
- how to position the 3D cursor




You'll also learn the fundamentals of:


- visibility layers
- adding objects
- deleting objects

18.1. The Viewport and its Contents

Aside from its header, the remainder of a 3D View window is its **viewport**. In Blender, you use viewports any time you need an up-to-date view of the scene you're working on.

Viewports are busy places. Go on a scavenger hunt and see what you can find in a simple viewport.

1. Launch Blender.
2. Just so we're all looking at the same scene, load the factory settings using *File* → *Load Factory Settings*.
3. Confirm *Erase All* with  (or .
4. If the NumLock indicator on your keyboard is unlit, press  so that numpad hotkeys will work properly.

(If you're unsure what  means, please review the Keystroke, Button, and Menu Notation module¹.)

You should see something like this:

¹ Chapter 9 on page 43

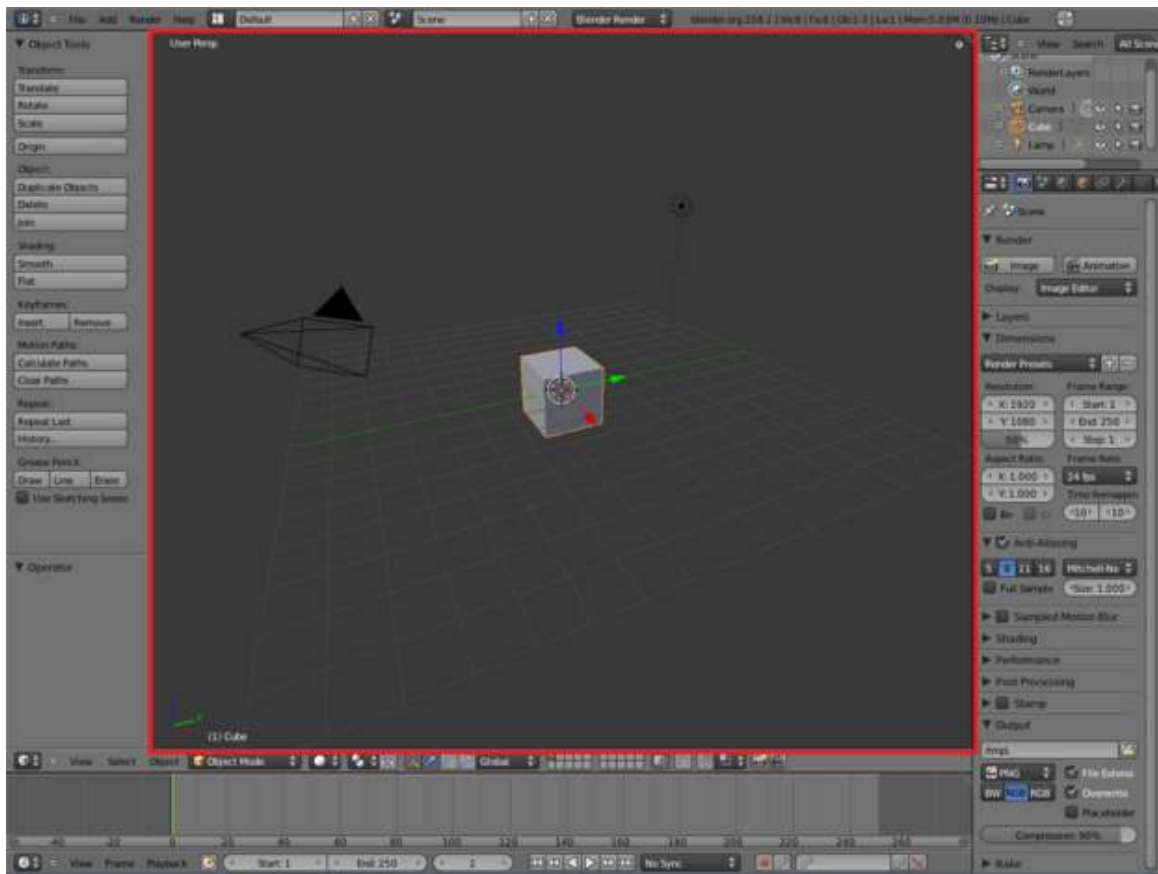
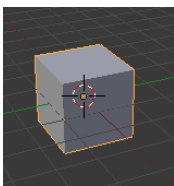


Figure 45

18.1.1. A Virtual Scavenger Hunt

Peer into the viewport at the default scene provided by the creators of Blender and find the following eight items:

In the Center



1. a solid gray cube with orange edges.

- This is the top face of the **default cube**, your first Blender object!



2. Three big arrows, one red, one green and one blue, their tails joined to a white circle

- This is not an object. It is the manipulator for the **3D transform widget**.
 - The arrows represent the directions of the **X**, **Z** and **Y** axes of the global coordinate system.
 - The circle represents the center of the selected object (namely, the cube).
 - If you don't know what a "global coordinate system" is, please review the module on Coordinate Spaces in Blender².
- It's possible that a tool is active. Press **[ESC]** to cancel any tool action.
 - Another possibility is that the manipulator has been disabled:
 - Toggle it on or off with **[CTRL] + [SPACE]** .

3.

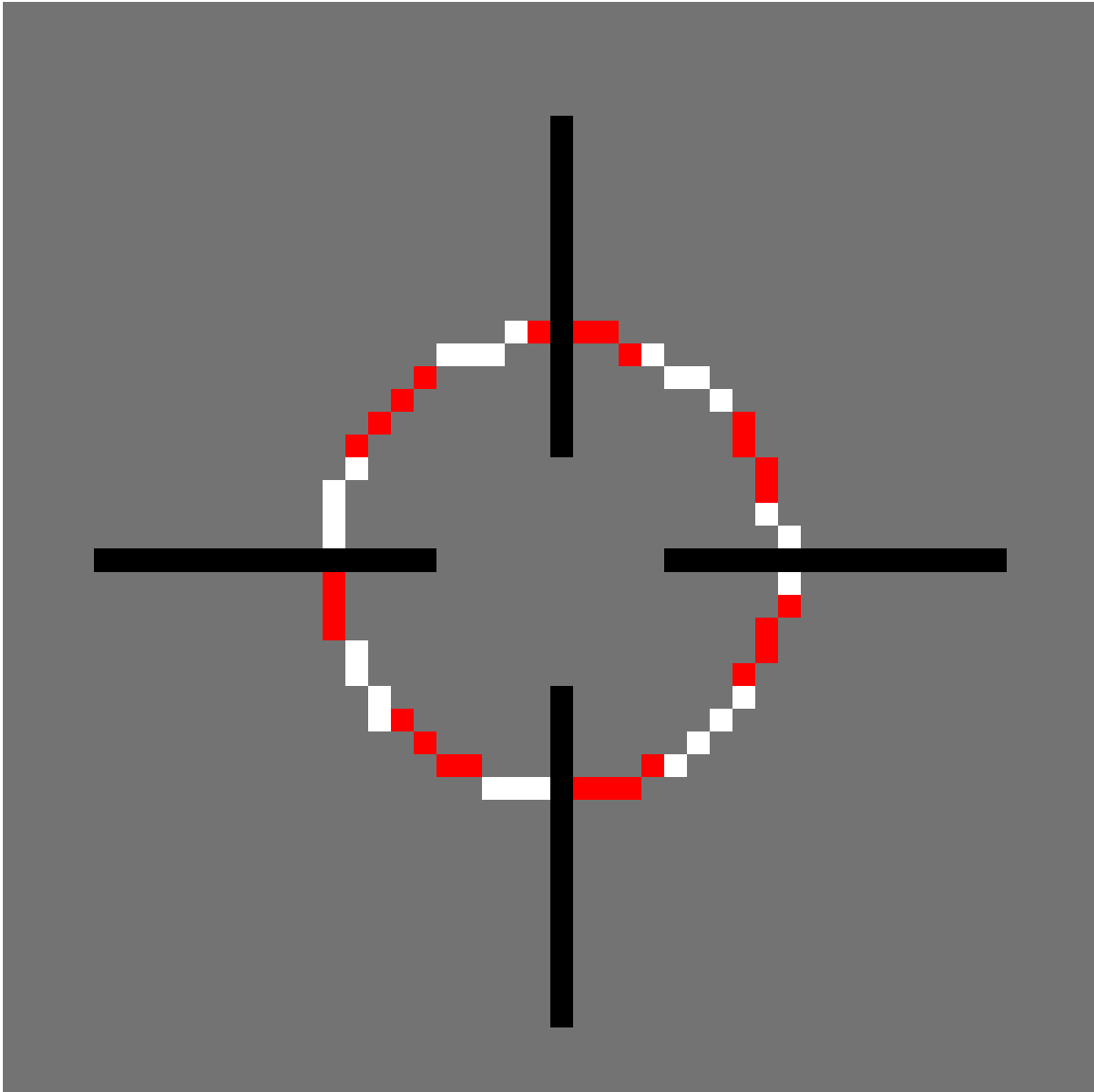


Figure 48

A red-and-white striped circle with black cross-hairs

- This is not an object. It is the **3D Cursor**, which indicates where newly-created objects will appear in the scene.
- The cursor is similar to the insertion point in a text editor, which indicates where newly-typed text will appear in a document.

In the Lower Left Corner

4.

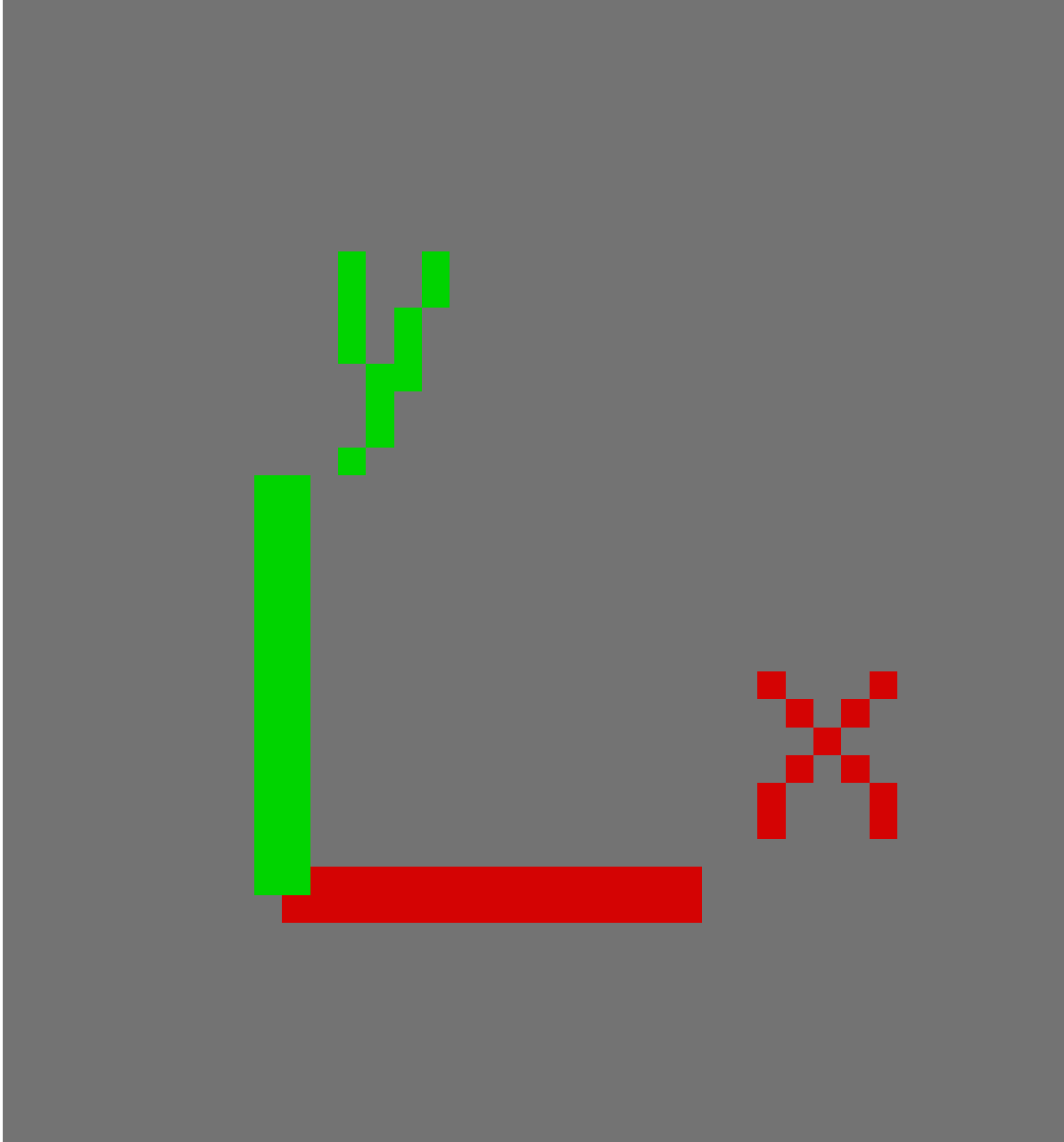


Figure 49

A red-and-green "L", with its arms labeled "x" and a "y"

- This is not an object. It is the **mini axis**.
- The arms of the "L" indicate the +X and +Y directions in the global coordinate system.

5. The notation "(1) Cube"

This is not an object. It is **object info**, indicating that:

- You're viewing the first frame of an animation.
- and
- The current or most recently selected object is named "Cube".

To the Right of Center

6.

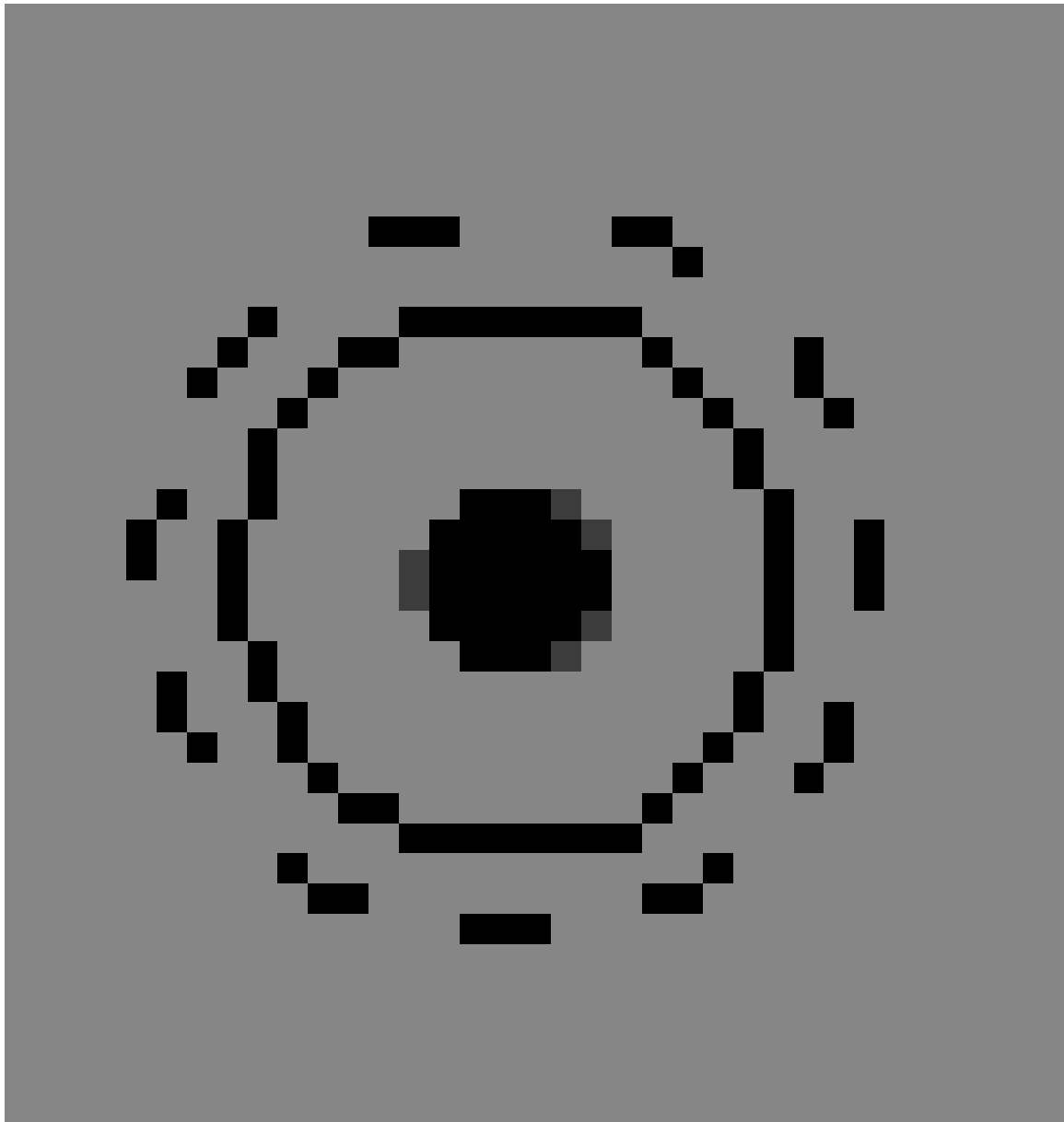
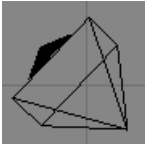


Figure 50

A black round thing that resembles a sun-symbol

This represents a **lamp**, a light source for the scene. (It is an object.)



7. A black triangular thing

This represents a **camera**, a viewpoint that can be used for rendering. (It, too, is an object.)

On a small display, the camera might initially lie outside of the viewport and thus be invisible. In that case, zoom out by scrolling with **[MMB]** until it becomes visible.

Throughout

8. A dark gray background, divided into **grid squares** by lighter lines.

Each grid square is one **blender unit** (or **BU**) on a side. A BU can be whatever you wish: an inch, a centimeter, a mile, or a cubit. Blender lets you decide what the scale should be for your scene in the Scene tab of the Properties Panel.

18.2. Modes

Blender has lots of **modes**, that is, settings that affect its behavior, and this is especially true of 3D View windows.

Sometimes it's not obvious what modes are active. This leads to **mode errors** where Blender will do something you didn't expect because you thought it was in one mode and it was actually in another.

In Blender, the function performed by a hotkey or mouse button can depend on:

- what mode the user interface is in,
- whether the keyboard is in NumLock mode,
- which window is active,
- the mode the active window is in,
- which item or items are selected,
- whether you've initiated a hotkey sequence.

It helps to recognize the common modes and get out of them.

18.2.1. Object Mode vs. Edit Mode

The 3D View windows are normally in Object Mode. In this mode:

- The mouse pointer is a white arrow.
- **[RMB]** is used to select objects in the scene

If there are objects in the scene, you can get into five other modes:

- Edit Mode: used to edit the shapes of objects
 - The mouse pointer is a thin inverse-video cross (it looks like a plus sign).

- **RMB** is used to select vertices, faces or edges of the current object.
- Press **TAB** to enter/exit this mode.
- Sculpt Mode
 - The mouse pointer is now a thin, orange circle.
- Vertex Paint
 - The mouse pointer is the same as in sculpt mode, a thin, orange circle.
- Texture Paint
 - The mouse pointer is a thin white circle.
- Weight Paint
 - The mouse pointer is again, a thin orange circle

These modes are also indicated by a menu in the 3D View header. You can use this menu to change modes.

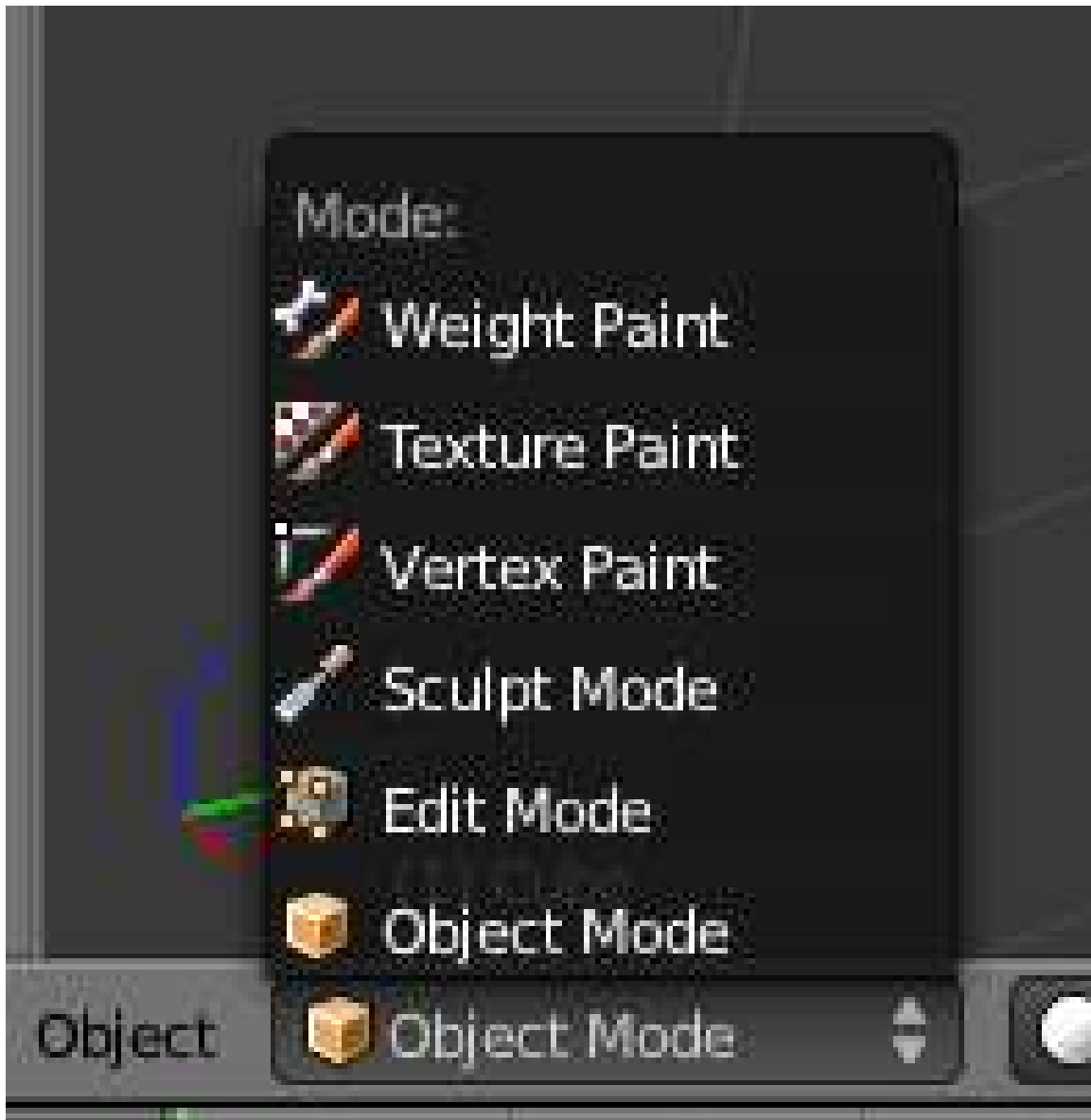


Figure 52

These modes are a setting shared by all 3D View windows. In other words, when you change the mode in one window, any other 3D View windows change mode also.


18.3. Viewport Options

Note:

The options in this section only affect 3D View viewports. They do not affect renders.

18.3.1. Solid vs. Wireframe


By default, the 3D View window draws objects using the Solid drawtype, in which surfaces are opaque. To toggle between Solid and Wireframe drawtype (edges only, no faces) for a particular viewport:

1. Activate the 3D View window.
2. Press .

Alternatively, you can choose these and other drawtypes from the "Draw type" menu in the 3D View window header.

18.3.2. Orthographic vs. Perspective

By default, viewports draw orthographic views. To toggle a viewport between orthographic and perspective views:

1. Activate the 3D View window.
2. Press .






(If you're unsure what the difference is, please review the "Orthographic Views" module and the "Perspective Views" module .)

... you should activate the View Name option. This is enabled by default and causes the name of the current view ("User Persp", for instance) to appear in the upper left corner of every viewport. If there is no text, then you can enable it by:

1. Accessing the User Preferences window.
2. Click on the *Interface* tab.
3. Enable *View Name*.

18.4. Undo

Since you've begun making changes to the scene, now is a good time to learn how to undo (and redo) changes.

- To undo a change, press  + .
- To redo a change you recently undid, press  +  + .

Blender puts a limit on how many steps you can undo. Instructions for changing this limit are in the "User Preferences Windows" module .

18.5. Changing Your Viewpoint, Part One

Each viewport has a **viewpoint**, which takes into account:

- the location of the viewer in the 3D scene (There doesn't need to be an object at that location.)
- the direction the viewer is looking
- the magnification (or zoom factor) used

Changing your viewpoint allows you to navigate your way through a 3D scene.

We'll start with three very basic techniques:

- Zooming
- Orbiting/View Rotation
- Perfect Views.

Additional techniques will be covered later in this module.

18.5.1. Zooming

Blender offers several ways to zoom in and out:

- Use **SCROLL**
- Click and drag vertically with **CTRL** + **MMB** .
- Use **NUM+** and **NUM-** to zoom in and out in small increments.

Note the following limitations of Blender's zoom feature:

- If the viewport is in orthographic mode, Blender zooms as if looking through a telescope. You can increase the magnification, but the viewpoint's location doesn't change. For this reason, you cannot zoom into or through objects in orthographic mode.
- If the viewport is in perspective mode, Blender zooms to a point—namely the center of the viewport. The viewpoint can pass through objects, but it can't pass beyond this point, no matter what you do. Zooming only gets slower and slower and slower. If the center of the viewport is somewhere you don't expect, zooming may appear to be broken.

18.5.2. Orbiting and View Rotation

Let's fly around in the default cube, viewing it from different angles. In this way you'll see that it really is a cube, centered on the origin, half above the X-Y plane and half below it.

1. Activate the 3D View window by placing the mouse pointer inside it.
2. Now you can:
 - Click and drag with **MMB** to orbit freely around the center of the view.
 - Use **SHIFT** + **ALT** + **SCROLL** to rotate the viewpoint vertically around the center of the view.

- Use **NUM2** and **NUM8** to rotate the viewpoint vertically around the center of the view in 15-degree increments.
- Use **CTRL** + **ALT** + **SCROLL** to rotate the viewpoint around the Z axis.
- Use **NUM4** and **NUM6** to rotate the viewpoint around the Z axis in 15-degree increments.

If this is all very confusing for you, don't worry! You'll learn as you get more experience.

When you are finished flying around the cube, you can restore the original view by reloading the factory settings with *File* → *Load Factory Settings*.

You may have pressed number keys above the letters instead of the ones on the numpad. If you do so, the default cube will vanish. This is because the scene consists of multiple layers, the default cube is in layer 1, and you've told Blender to move the selected object (the cube in this case) to the layer of the number you just pressed. For instance, **2KEY** tells Blender to move the selected object to layer 2. To move it to layer 1 again, press **1KEY**. You can view the different layers by clicking on the little squares on the layer map:



Figure 53

Note:

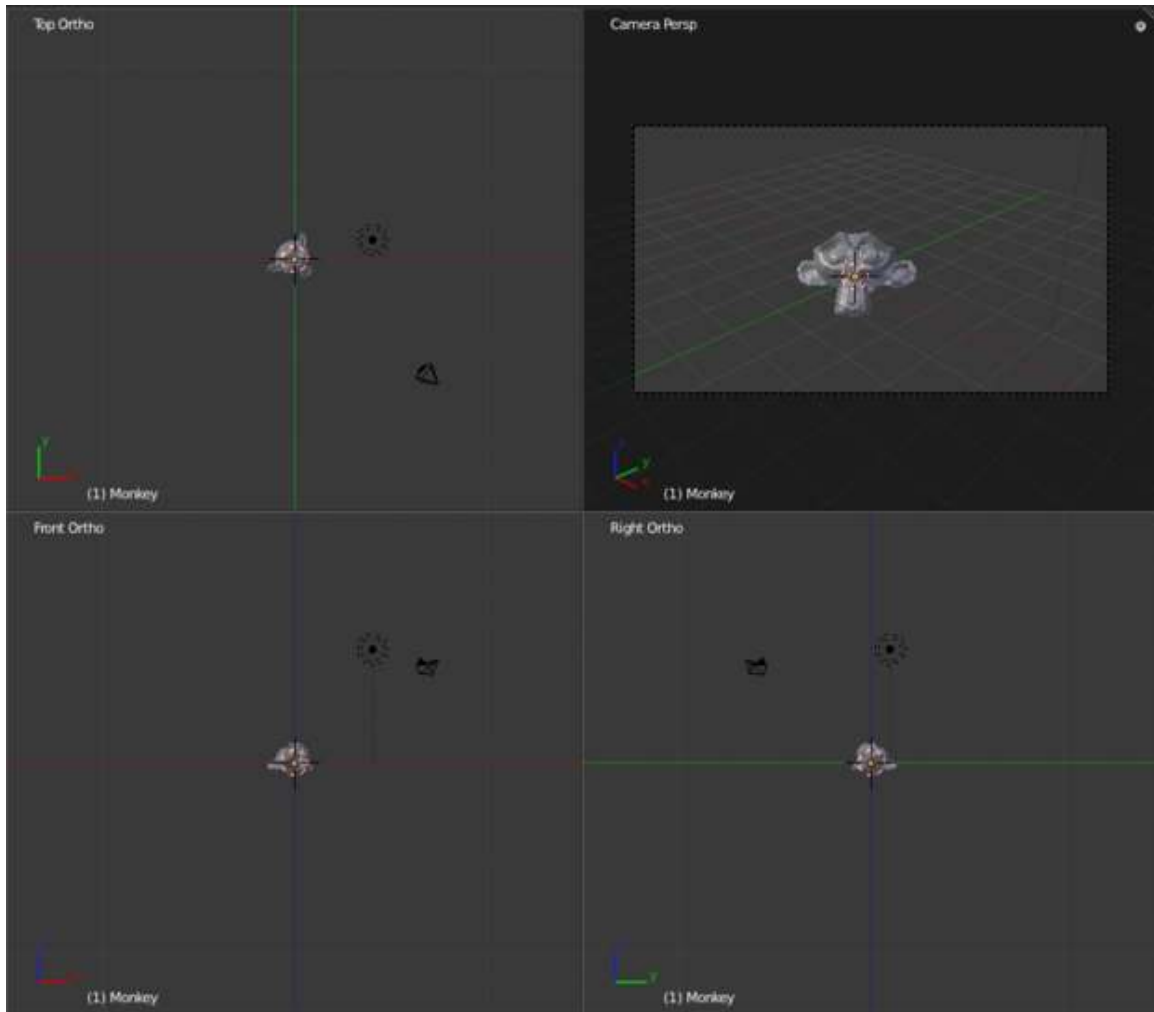
Because the center of the viewport is invisible (not marked in any way), it's hard to tell where it is. This sometimes results in unexpected behavior during rotation.

18.5.3. Perfect Views

It's often useful to get a **perfect view** of a scene—in other words, to view it along one of the main axes, with the other two main axes oriented up-down and left-right.

Perfect View Hotkeys			
Hotkey	View	Axis Pointing Right	Axis Pointing Up
NUM7	"top"	+X	+Y
CTRL + NUM7	"bottom"	+X	-Y
NUM1	"front"	+X	+Z
CTRL + NUM1	"rear"	-X	+Z
NUM3	"right side"	+Y	+Z
CTRL + NUM3	"left side"	-Y	+Z

The following screenshot shows all three perfect views plus camera perspective for the Suzanne primitive:

**Figure 54**

18.6. Positioning the 3D Cursor

Positioning the 3D cursor is a very basic operation, yet one that many beginners find challenging. It touches on an issue common to all 3D graphics software: "How do you specify points in a 3D scene when we can only see two dimensions at a time?"

18.6.1. Basic Technique

1. Go into either Object Mode or Edit Mode.
2. Move the mouse pointer to the desired position (in any viewport).
3. Click .

 **Warning**

This technique will fail if the 3D manipulation widget is enabled and your desired position is too close to it. Clicking **LMB** on or near the widget (the white circle with the colored arrows) will initiate a transform operation; the object's outline will turn white and the mouse pointer will begin dragging the object around. If this happens, press **ESC** to cancel the transform operation.

Warning

Clicking **LMB** in a viewport can only reposition the cursor in two out of three dimensions. (The cursor's projected distance along the central line-of-sight remains unchanged.) For this reason, any time you reposition the cursor this way you should immediately verify its position using a different viewpoint.

18.6.2. Two Challenges

Challenge #1. Using only tools presented thus far, try positioning the 3D cursor on the virtual camera.

Try it!

When you're done, check your work by orbiting the camera.

Perhaps you thought you were done when you clicked on the camera. But the moment you changed your viewpoint, you probably found that the 3D cursor was actually behind (or in front of) the camera.

Hints:

- Try positioning the cursor in two different perfect views.
- Use orthographic views, not perspective ones.

Challenge #2. Using only tools presented thus far, try repositioning the 3D cursor at the origin (that is, at the center of the cube).

As before, check your work by orbiting the cube. Don't spend too much time on this.

"I found that I would select the cube when left clicking on it in object mode, if the **"Use 3d transform manipulator"** button was enabled. To toggle this off, you click on the **gray pointing hand** in the 3d panel header, or (Ctrl Space)."

"When you want the cursor back into the cube, just select the camera with **RMB**, put the cursor into the cube following the steps above, and re-select the cube with **RMB**."

"I've discovered it helps a lot if you are in Object Mode and not in Edit Mode. I wrote the following before discovering this: *The problem with this exercise, for me, is that left clicking on the cube selects the cube instead of moving the 3d cursor. If I click on the cube outside of its central white circle I can get the cursor to move there, but only to outside of this white circle, and even then this only works sometimes.*"

"I failed at this until I had zoomed in close enough to the cube. When I was too far zoomed out I kept selecting the cube rather than creating an edit point."

"I had the same problem and found it was because the cube was selected. I made sure I was in object mode, right clicked on the camera to select the camera instead of the cube, and I could then position the edit point in the cube. However, doing this messed up the next part of the tutorial because you cannot switch into edit mode with the camera selected! Perhaps the suggestion of trying to put the 3D cursor in the cube should be dropped as it raises too many questions at this stage."

"You can deselect all by pressing the AKEY or the select button in the 3D View."

"Use wireframe mode works better to get the cursor in."

"To get it back in the cube: 1) Make sure you're in object mode. 2) Select the cube. 3) Object > Snap > Cursor to selection (cursor refers to the 3D cursor here) so it puts it right in the middle of the cube."

"I think it's an essential point to note that *in order to place the cursor inside the cube, the cube must NOT be selected*. AKEY was probably the best way to deselect the object."

"If I remember correctly, undo history gets cleared when you switch between object and edit mode."

"I wasted a lot of time here. *Thank you* to the reader who suggested (on the 3D view header) Object > Snap > Cursor to selection. It was the only thing that worked to get the cursor visible again and placed where clicked."

"I missed the point of the exercise first time around. You can't set a 3D point on a 2D screen without technique. Orthographic views are crucial. I am just learning, but take that, at least, away from it."

18.6.3. More Ways to Position the Cursor

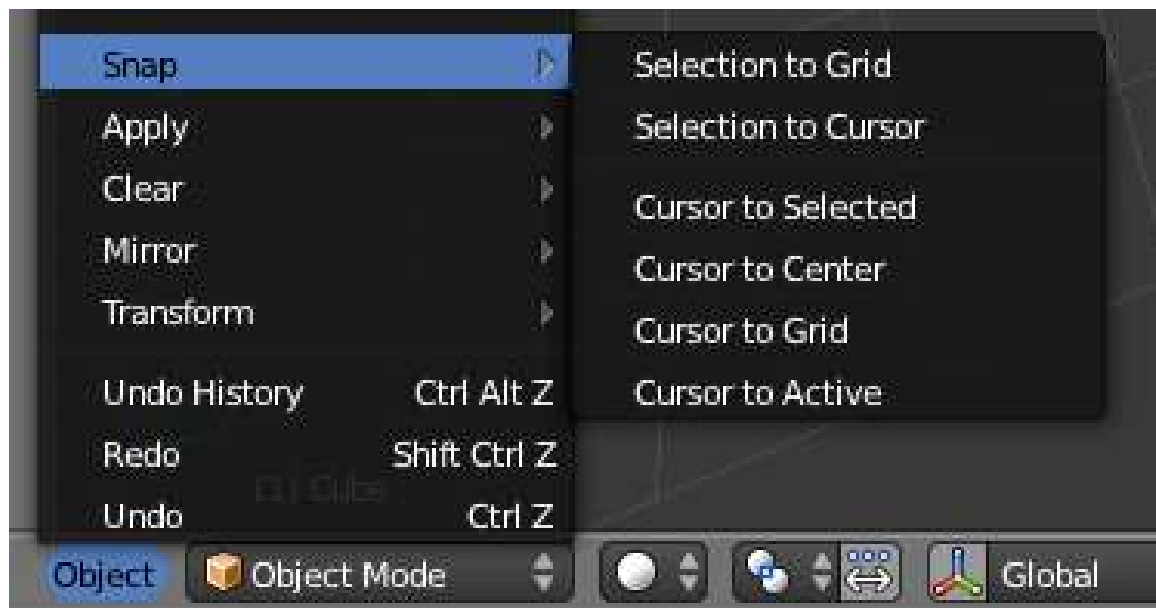


Figure 55

Here's an easy way to position the cursor at the center of an object:

1. Make sure Blender is in Object Mode, with the object selected.
2. Move the mouse pointer to any 3D View window.

3. Snap the cursor to the selected object using either:

- **[SHIFT]** + **[S]** *Cursor to Selected*
- or
- *Object* → *Snap* → *Cursor to Selected*

Here's 2 easy ways to relocate the cursor to the scene's origin (0, 0, 0):

1. Move the mouse pointer to any 3D View window.
2. Press **[SHIFT]** + **[C]** to reset the cursor to the origin.
 - Note that this also changes the view location, meaning that when you zoom in, you won't zoom in to the scene origin.
3. A better way is to click *Object* → *Snap* → *Cursor to Center*
 - You can also do this by **[SHIFT]** + **[S]** → *Cursor to Center*.

18.7. Changing Your Viewpoint, Part Two

Now you'll learn some additional techniques for obtaining the view you want:

- Dollying
- Centering
- Jumping to the camera's viewpoint
- Zooming in on a selected area

18.7.1. Dollying

When you orbited the cube, the viewpoint's position and direction both changed at the same time. You also can shift the viewpoint up-down or left-right *without* changing its direction. (This is similar to the side-scrolling effect in the classic Mario and Sonic video games.)

This is called "dollying", and it's an important skill to master. Try it now:

1. Activate a 3D View window by placing the mouse pointer inside it.
2. Now you can:
 - Use **[SHIFT]** + **[SCROLL]** to dolly up and down.
 - Use **[CTRL]** + **[NUM2]** and **[CTRL]** + **[NUM8]** to dolly up and down in small increments.
 - Use **[CTRL]** + **[SCROLL]** to dolly left and right.
 - Use **[CTRL]** + **[NUM4]** and **[CTRL]** + **[NUM6]** to dolly left and right in small increments.
 - Click and drag with **[SHIFT]** + **[MMB]** or **[SHIFT]** + **[ALT]** + **[LMB]** to dolly freely in the viewplane.

You will likely find this to be a distraction in some cases. To move the viewpoint position back to the center, snap the cursor to the center, then click *View* → *Align View* → *Center View to Cursor*. You could also snap the cursor to the center then press **[CTRL]** + **[NUM.]**.

18.7.2. Centering

When you zoom or rotate the view, you always zoom or rotate around the center of the view.

To center the view on an arbitrary point:

1. Move the 3D cursor to the point of interest.
2. Verify the cursor position from a second viewpoint.
3. Press **CTRL** + **NUM.** to center the view.

To center the view on an object in the scene:

1. Make sure Blender is in Object Mode.
2. Zoom out until the object is in the viewport.
3. If any objects are selected, use **A** (or *Select* → *Select/Deselect All*) to deselect them.
4. Select the object of interest by clicking **RMB** on it.
5. Press **NUM.** to center the view.

18.7.3. Jumping to the Camera's Viewpoint

To see the scene as the virtual camera sees it, press **NUM0**. Afterwards, you can rotate, dolly, and zoom normally, but the virtual camera will not follow. To go back to your previous view, press **NUM0** again. (In the latest versions of Blender, the virtual camera can be made to follow all the changes made in viewpoint while in camera view by checking the option "Lock Camera to View" on the Transform panel. Hit "N" on your keyboard to bring the transform panel. To disable this option you can uncheck that and get back to previous settings.)

18.7.4. Zooming into a Selected Area

Suppose you want to get an extreme closeup of a particular area. Because there's no center mark on the viewport, you might have to dolly and zoom several times to get the desired view.

The shortcut for zooming to an area is:

1. Activate a 3D view window that contains the area of interest.
2. Press **SHIFT** + **B**. A crosshair appears in the viewport.
3. Click and drag with **LMB** to draw a rectangle around the area of interest.
4. When you release **LMB**, the viewport will zoom in on the area you selected.

18.8. Visibility Layers

Every object in the scene is assigned to one of 20 **visibility layers**.

Visibility layers have many uses:

- You can put scenery, characters, particles, and lamps in different layers, to help organize your scene.
- By changing which layers are visible, you can simplify your view of the scene and work with only one or two layers at a time.
- When rendering, only visible layers are included. You can use this to render your scene layer by layer, checking each layer separately.
- You can configure lamps to illuminate only objects in the same layer.

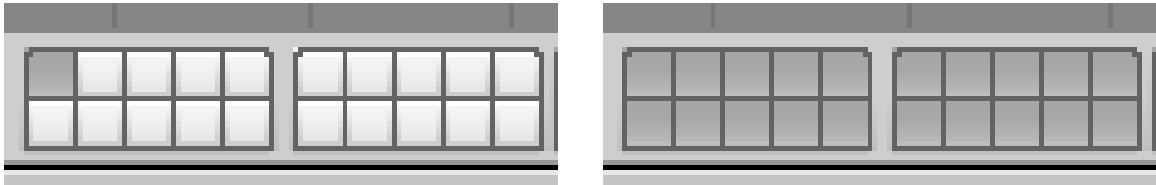


Figure 56 Left: Viewing a layer 1 only.
Right: Viewing all 20 layers.

In Object Mode, you can tell which layers are visible by looking at the twenty small boxes located in the 3D View header between the Transform Orientation menu and the "Lock" button. The top row of boxes represents layers 1 through 10, with 1 being the leftmost and 10 being the rightmost. Similarly, the bottom row of boxes represents layers 11 through 20.

18.8.1. Hotkeys

- To view layer 1 only, press .
- To view layer 2 only, press .
- To view layer 10 only, press .
- To view layer 11 only, press + .
- To view layer 19 only, press + .
- To view layer 20 only, press + .

Note:

The hotkeys in this section *will not work* if you've enabled numpad emulation in the User Preferences window. See the "User Preferences Windows" module for more details.

On the AZERTY keyboard layout, the standard number keys are the &é"(-è_çà keys. Do not use unless you want to toggle visibility as explained below.

Holding down while selecting a layer (by keyboard or mouse) will, instead of making only that layer visible, toggle the visibility. In this way, you can select combinations or to hide particular layers.

The key to press to select all layers at once differs by keyboard layout. It is:

- (the key under Esc) on UK keyboards,
- on US ones,
- on German, Swedish and Finnish ones,
- on Danish ones,
- on AZERTY ones,
- on Norwegian ones,
- on Brazilian Portuguese ones, and
- on Italian ones.

After pressing the aforementioned key, holding down while pressing it again will restore the visibility settings you had *before* you made all layers visible.

When only one layer is selected, new objects are automatically assigned to that layer. When two or more layers are visible, new objects are assigned to the layer that became visible most recently.

To move one or more objects to a different layer:

1. Select the object(s).
2. Press **M**.
3. Choose the new layer from the pop-up.

18.9. Adding and Deleting Objects

To delete the default cube from the scene:

1. Choose *Object Mode* from Mode menu in the 3D View header.
2. If the cube does not have orange edges, then it is **deselected**. To reselect the cube, move the mouse pointer onto the cube and click **RMB** (or **CMD** + **LMB**).
3. Press **X** or **DELETE** to delete the selected object.
4. Confirm *Delete* with **LMB** (or **ENTER**).

To add a new monkey head (or any other primitive) to the scene:

1. Position the 3D cursor where you want the head to go.
2. Press **SHIFT** + **A**.
3. Choose *Mesh* → *Monkey*.

In the Blender world, this model is known as Suzanne.

Suzanne will be created facing in the global +Z direction. But if you select *Align to View* in the Operator panel (below the Tool Shelf) then Suzanne and any other object will face directly into the viewport.

18.10. Count Your Polys

If you want to count the polygons in your scene, this data is available in the Info Header.

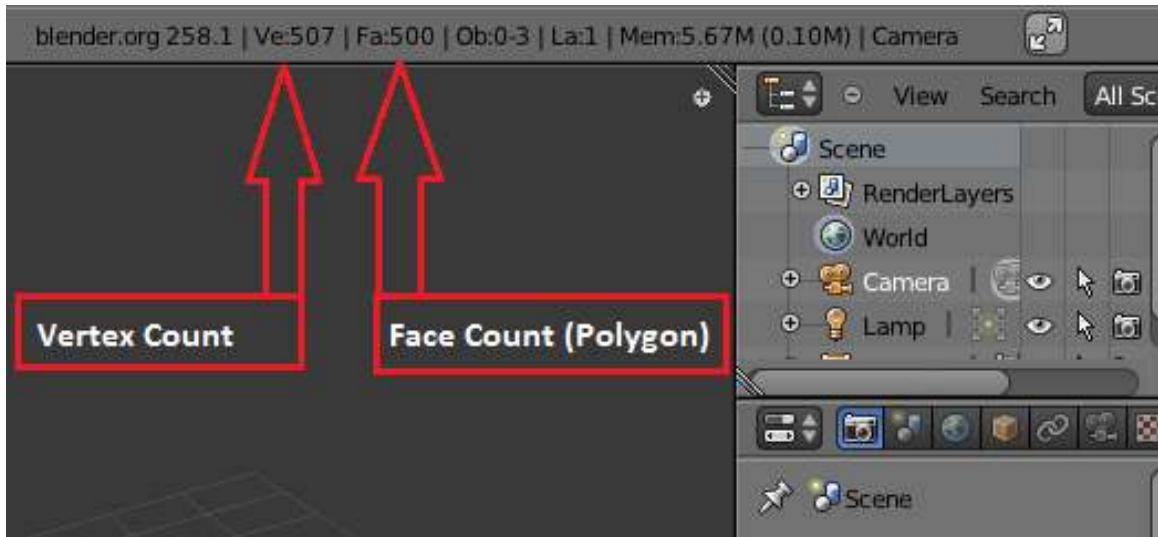


Figure 57

As you can see in the above image, this scene has 507 vertices and 500 faces (polygons).

19. Blender Interface (version 2.5)

This is the revised interface in **Blender 2.5x**. If Blender 2.49 looks like something out of *Star Trek*, then the new Blender must be *Star Trek: The Next Generation!*

The major interface revision was initially resisted by many experienced blender users; but it has turned out that noobs are able to find things easier. The updates have also brought it much closer to on par with mainstream 3D content creation suites.

For the rationale behind the redesign, read [this](#)¹.

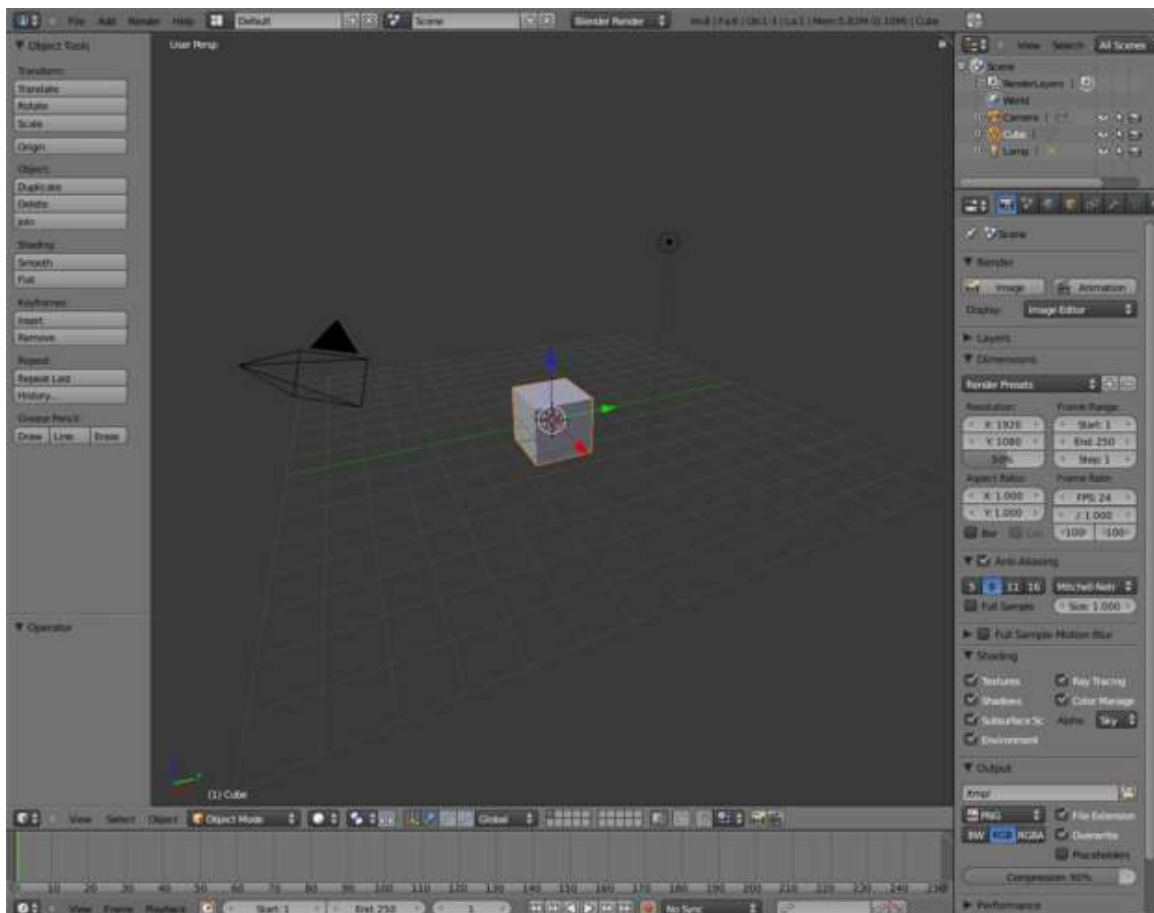


Figure 58

¹ http://download.blender.org/documentation/bc2008/evolution_of_blennders_ui.pdf

20. Blender Windowing System (version 2.5)

20.1. The Blender Workspace

Blender prefers to run full-screen, because it manages its own windowing system. This divides the screen up into rectangular, non-overlapping *windows* (sometimes called *areas*). These are marked out with coloured borders in the following screenshot of the default Blender 2.55 workspace layout:

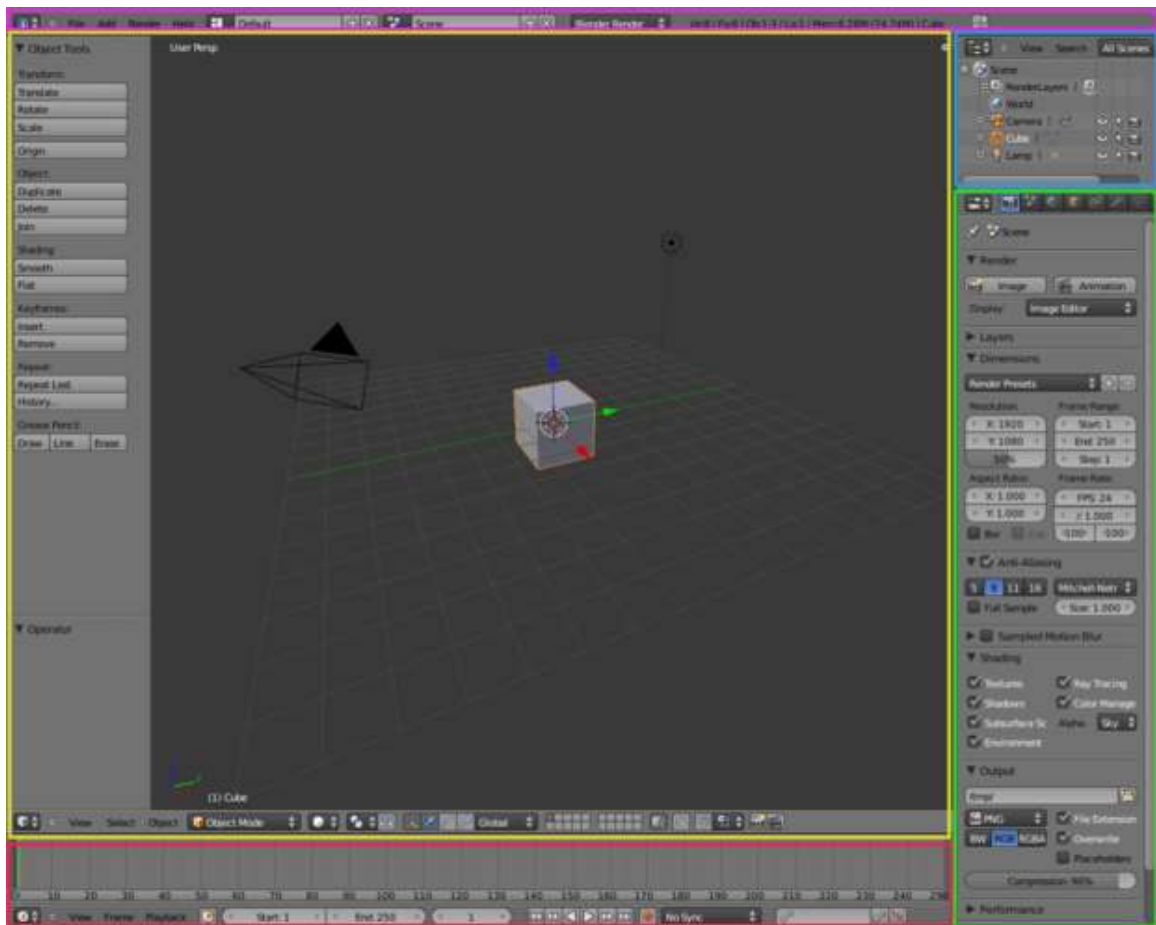


Figure 59

Each window has a *header*, which always shows an icon at its leftmost end which identifies the contents of the window. This icon has a pair of upward- and downward-pointing triangles next to it. If you click with **[LMB]** on this icon, a menu pops up which allows you to change what the window is showing:

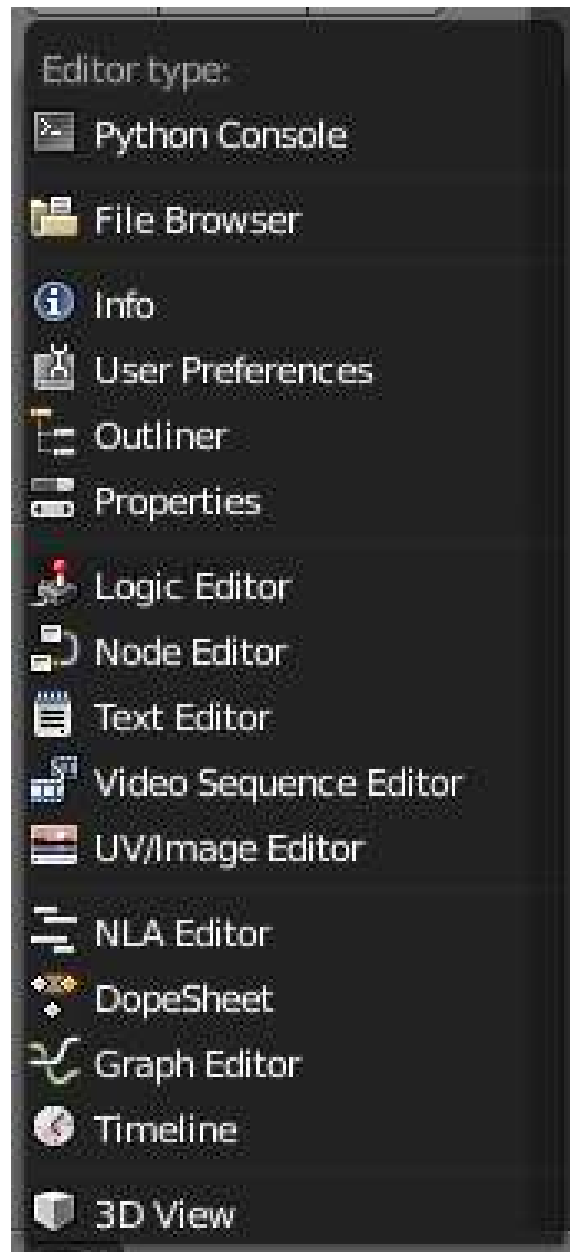


Figure 60

The header can be at the top or bottom of the window. If you click with **RMB** on the header, a menu pops up which lets you move the header (to the top if it's at the bottom, or vice versa), or maximize the window to fill the entire workspace:



Figure 61

To hide the header completely, move the mouse to the edge of the header furthest from the edge of the window (i.e. the top edge of the header if it is at the bottom of the window, or vice versa); it will change into a vertical double-headed arrow. Now click with **LMB** and drag towards the window edge, and the header will disappear. In its place, you will see the following symbol appear at the corner of the window:

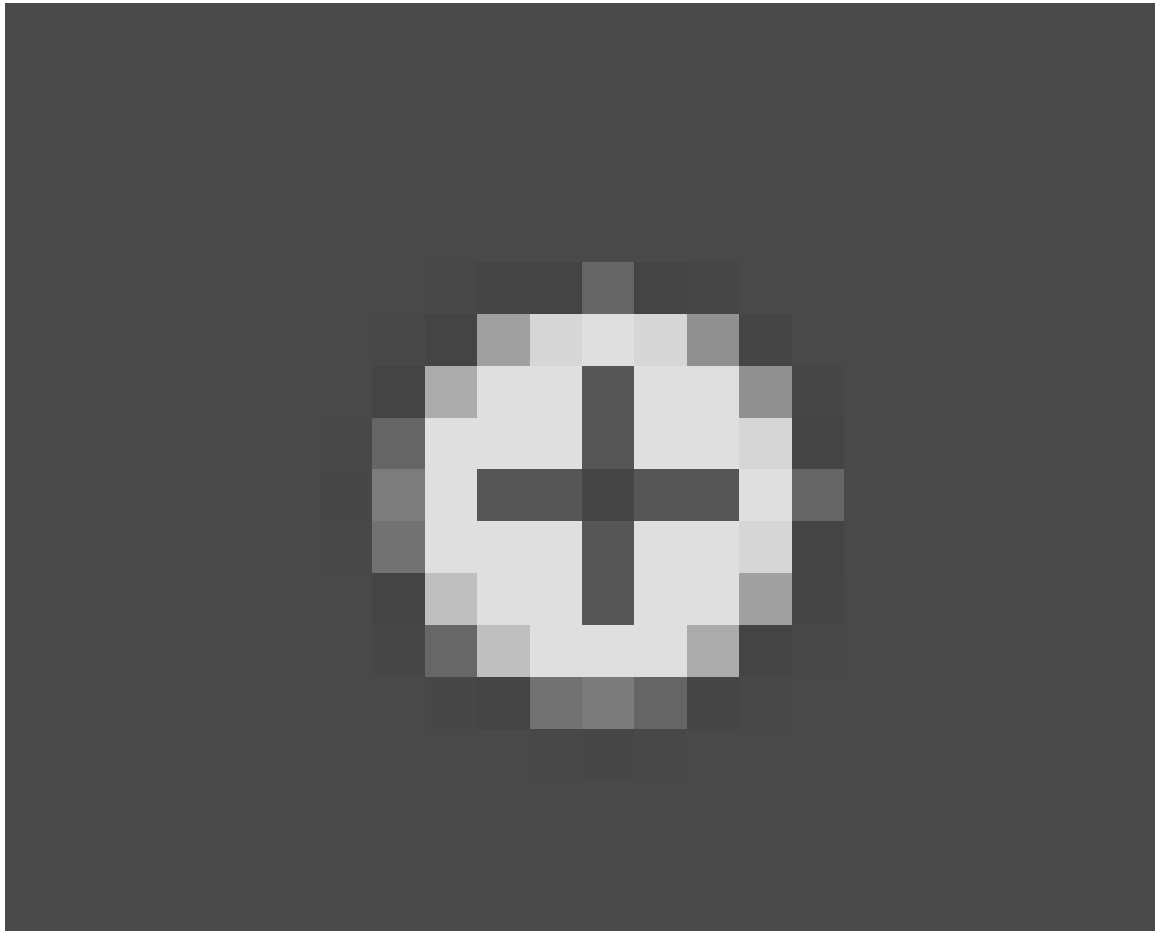



Figure 62

. Click this with  to bring the header back.

Note:

Any window can be changed to any type. You can have multiple windows of the same type.

Note:

The workspace layout is saved along with the document. Anybody subsequently opening the document will see the last-saved layout.


20.2. The Active Window

The **active window** is the one that will respond if you press a key. Exactly one Blender window is active at any given time; this is the one containing the mouse pointer. (When a hotkey fails to work as expected, it's often because the mouse pointer has strayed into a neighboring window.) To change the active window, simply move the mouse pointer into the window you wish to activate.

Note:

When a window becomes active, its header gets brighter.

20.3. Resizing A Window

When you move the mouse to the boundary between two windows, the pointer changes to a double-headed arrow. The arrows indicate the directions in which you can drag the boundary with , to resize the adjoining windows accordingly.

20.4. Shelves

You will notice that the 3D View window (the largest window in the above screenshot, outlined in yellow) has a bunch of buttons down the left. This rectangular portion is called the *Tool Shelf*. This is like a window within a window, and you can drag the boundary between it and the main part of the 3D View to resize.

If you drag all the way to the window boundary, the shelf will disappear. In its place, the following symbol will appear:

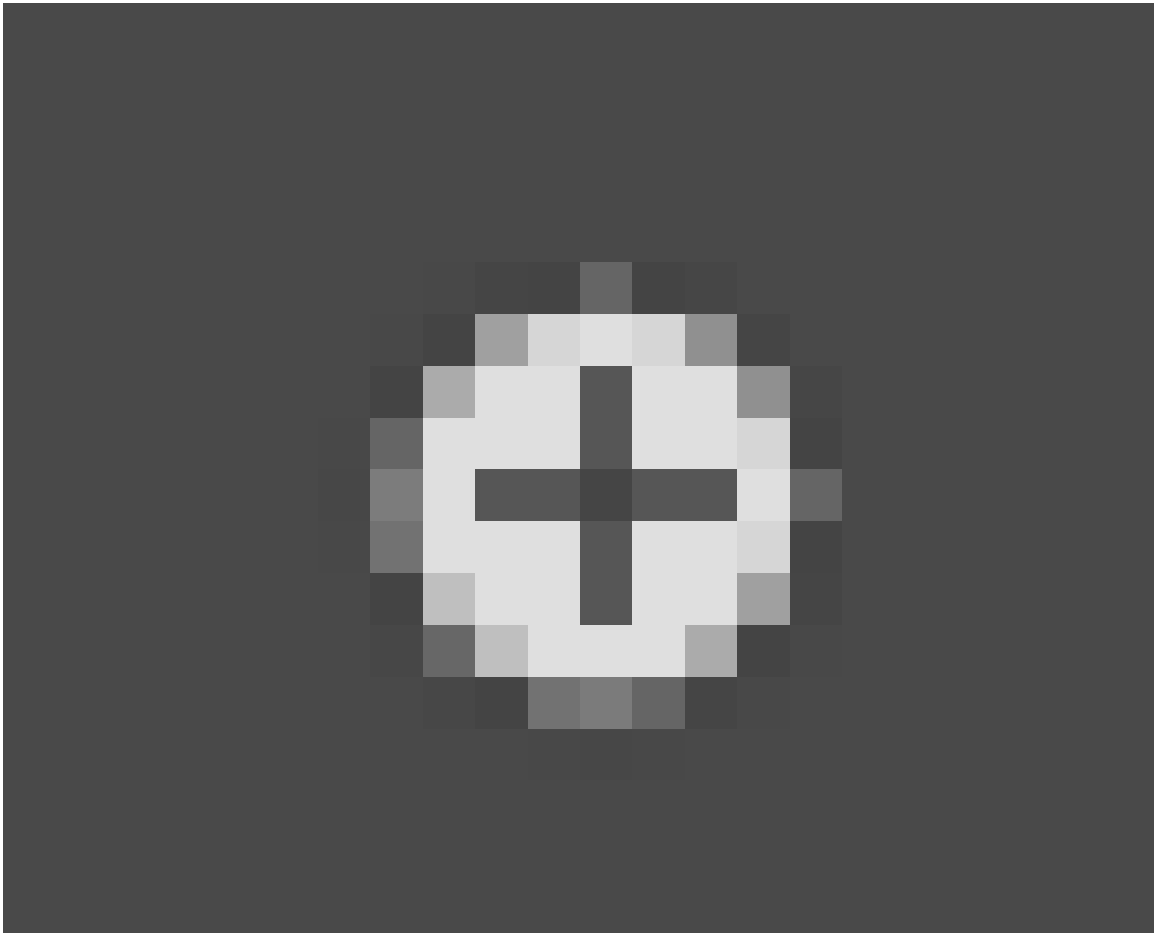


Figure 63

. Click that to bring the shelf back

20.5. Too Much To Fit

If a window or shelf contains too much information to fit within its display area, scrollbars will appear along the bottom or right edge. You can scroll the contents by dragging these with **[LMB]**; alternatively you can drag with **[MMB]** directly within the contents.

A window header may also contain more than fits within its displayable area. There is no explicit visual clue for this (except maybe the tell-tale of widgets being cut off at the right edge), but if it happens, you can drag sideways within the header with **[MMB]** to scroll its contents.

Note:

Wraparound Mouse: If, while scrolling by dragging with **[MMB]**, the mouse reaches the edge of the display area, it will “magically” disappear and reappear on the opposite edge. This lets you keep on scrolling without having to let go and reposition.

20.6. Panels

Within certain windows, you will see occurrences of the following symbols



Figure 64

with headings next to them: click on the rightward-pointing one to make the panel appear, whereupon the symbol is replaced with the downward-pointing one; click that to make the panel disappear, and change the symbol back to the rightward-pointing one.

20.7. Splitting And Joining Windows

At the top right and bottom left of every window, you will see something like this:

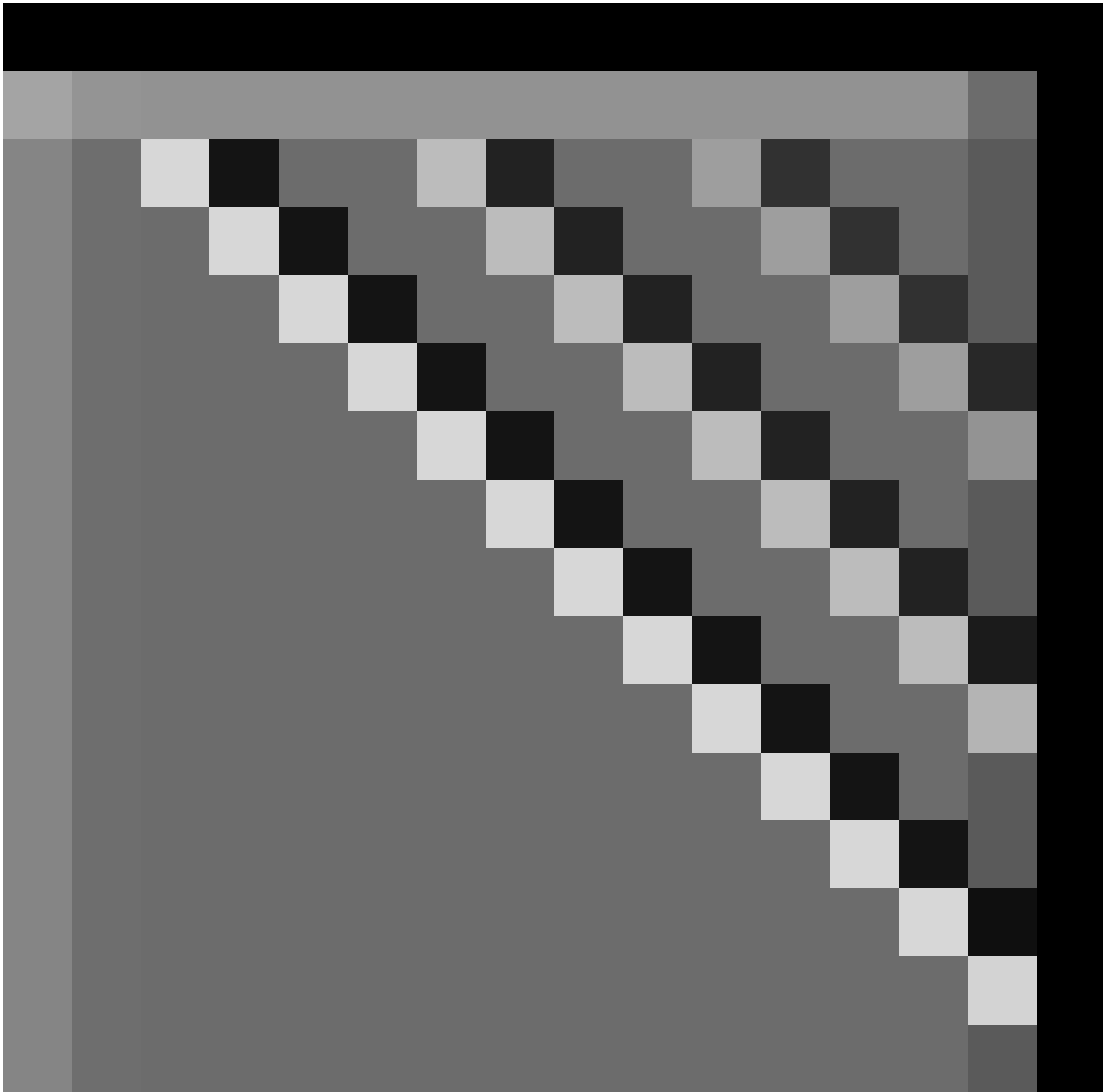


Figure 65

. If you move the mouse over this icon, you will see the pointer turn into a cross. At this point, you can do one of the following by clicking and dragging with **[LMB]** :

- Split the window into two copies horizontally by dragging horizontally away from the edge.
- Split the window into two copies vertically by dragging vertically away from the edge.
- Join the window to the adjacent one horizontally (getting rid of it and taking over its space) by dragging towards it.
- Join the window to the adjacent one vertically (getting rid of it and taking over its space) by dragging towards it.

Of course, the last two are only possible if there *is* in fact another window in that direction. Note also that you can only join windows horizontally that are the same height, and windows vertically that are the same width.


20.8. The Default Workspace

If you look at the above screenshot of the default workspace, you will see the following window types:

- The menu bar at the top (outlined in purple) is actually a window, called Info. In previous versions of Blender, you could resize this to reveal the User Preferences, but in 2.5x they have been moved to their own window type. Instead, all you can see here if you enlarge the window are some debug messages, which will probably be removed once this version of Blender goes to final release.
- The largest window on the screen (outlined in yellow) is the 3D View. This is where you work on your model.
- The Properties window is outlined in green; this is where most of the functions are located for performing operations on models, materials etc. In previous versions of Blender this was called the Buttons window, and was degenerating into a disorganized grab bag of functionality that made it difficult to find things. This has been cleaned up in a major way in 2.5x. Note also that it defaults to a vertical layout, rather than the horizontal one of previous versions. The new design prefers a vertical layout, which works better on today's widescreen monitors.
- The Outliner (in blue) gives you an overview of the objects in your document. As your models get more complex, you will start to appreciate the ability to find things quickly here.
- The Timeline (in red) becomes important when you're doing animations.

Of course, this is just the default layout. For example, if you're doing a static model or scene, not an animation, you can get rid of the Timeline. If you're doing heavy script development, you'll probably want the Console available to hand to try things out. And so on.

20.9. Workspace Presets

Up in that Info window/titlebar, you will see a menu with an icon looking like four little squares. Clicking on it with  will show the following menu:

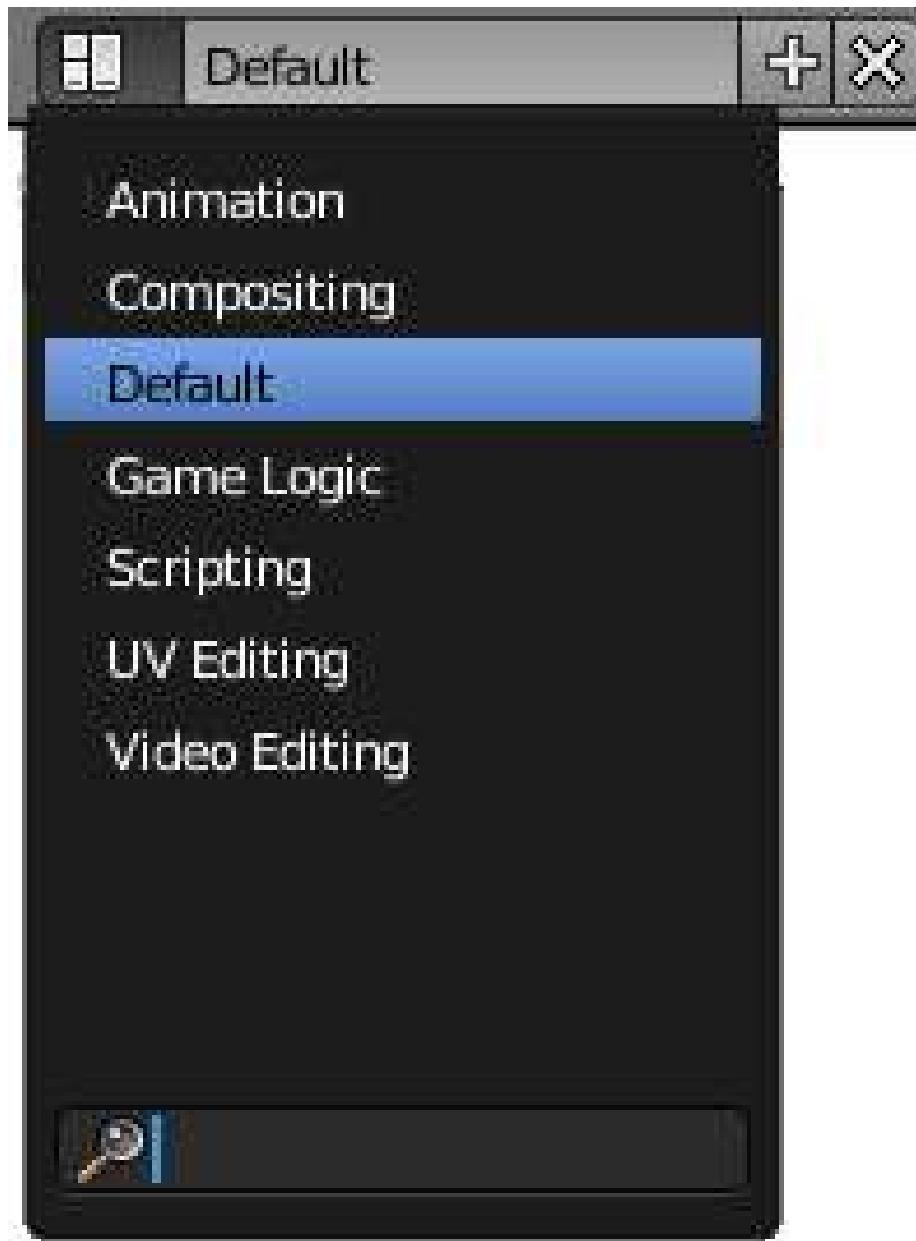


Figure 66

Selecting from this menu lets you quickly switch between various predefined workspace layouts, tailored to various workflows. Try it and see—you can return to the default layout by selecting “Default” (but note that any changes you make to the layout are immediately associated with the name being displayed here). The menu even has a search box at the bottom: typing some text in here will restrict the menu to only showing items containing that text. Hard to believe it has much use here, but perhaps in complicated projects which need dozens of different layouts, a search function could become very useful indeed!

Note also the “+” and “X” icons to the right of the menu; clicking “+” creates a new entry which is a duplicate of the last-selected entry, while clicking “X” gets rid of the currently-selected entry.

You will see these conventions appear consistently in menus elsewhere in Blender's new, revamped interface.

21. User Preferences Window (version 2.5)

21.1. Saving User Preferences

Most other applications have a place to keep user-configured settings (including document defaults), separate from any actual documents the user creates. Blender works in a slightly different fashion: all your user-configured settings are saved in *every* document you create. Each time you create a new Blender document, it actually just reloads your *default* document, which is called startup.blend. To save your current state as the default document, press **CTRL** + **U**. This will save *everything* you've done to the current in-memory document, including objects and materials created, to startup.blend. If for some reason you want to restore the default settings to how they were when you downloaded Blender, you can either delete that file, or click *Load Factory Settings* in the File menu.

21.2. Accessing User Preferences

You can open your preferences by selecting "user preferences" from the file menu. When you bring it up, you will see a row of buttons across the top:



Figure 67

These buttons select which section of your user preferences to display.

You can also choose it from the window type menu in any convenient window, since the User Preferences is a window type like any other. But if you save your preferences in this state, the next time you create a new document, it will have the preferences window open... probably not what you want.

21.3. Configuring Your Preferences

This tutorial only covers a couple of essential preferences, so you are able to get to modelling and other fun quicker. Feel free to browse through and tryout some of the preferences though, you can always restore the preferences to the factory setting if needed.

21.3.1. Auto Save

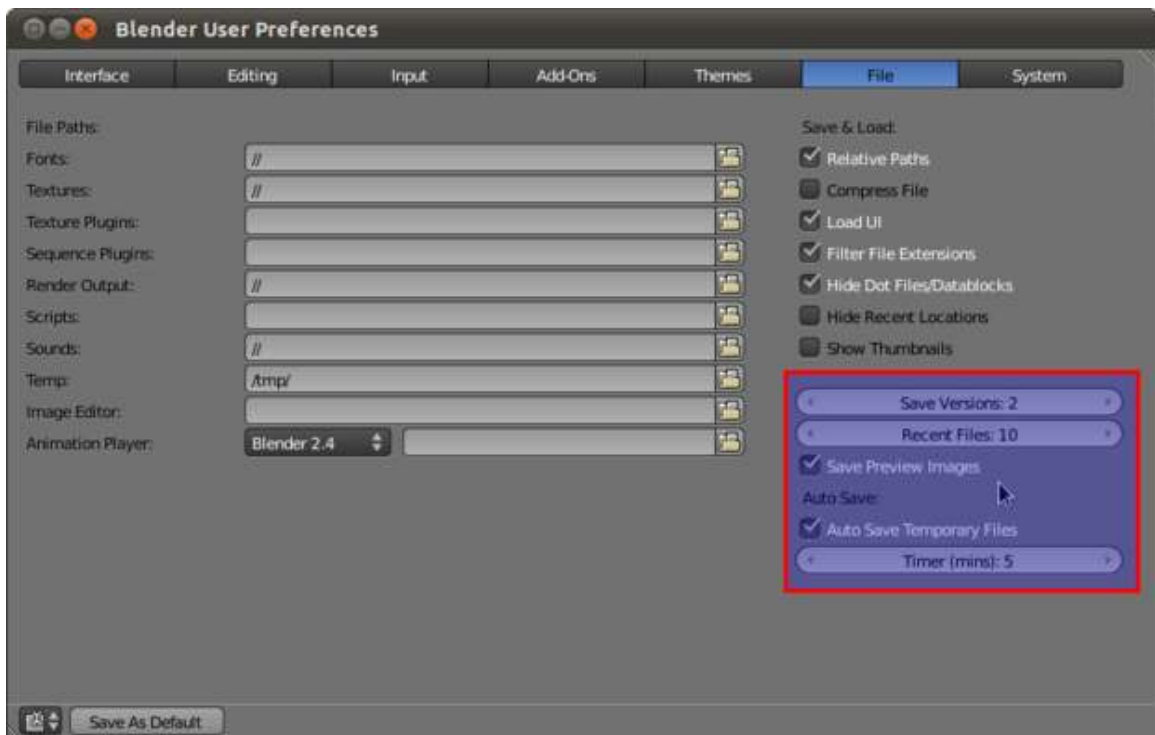


Figure 68

Here we have the Auto Save options in Blender. They are on the first screen you see in the user preferences window by default, under the *File* tab. Here's a brief description of each control:

- Auto Save Temporary Files - Toggle checkbox to control whether or not Blender automatically saves.
- Save Preview Images -
- Save Versions - This selection controls how many copies of one file Blender should save.
- Recent Files - This selection controls how many files total Blender is allowed to keep
- Timer - This selection controls how often files are auto-saved.

21.3.2. Number of Undo Levels



Figure 69

This is the Undo options. These options are located under the *Editing* tab in the user preferences window. Brief descriptions of the controls:

- Global Undo -
- Steps - This selection simply controls the number of times you can undo. How many is dependent on your system memory. For example, if you have 4 GB you might consider increasing this number.
- Memory Limit -

21.3.3. Numpad Emulation

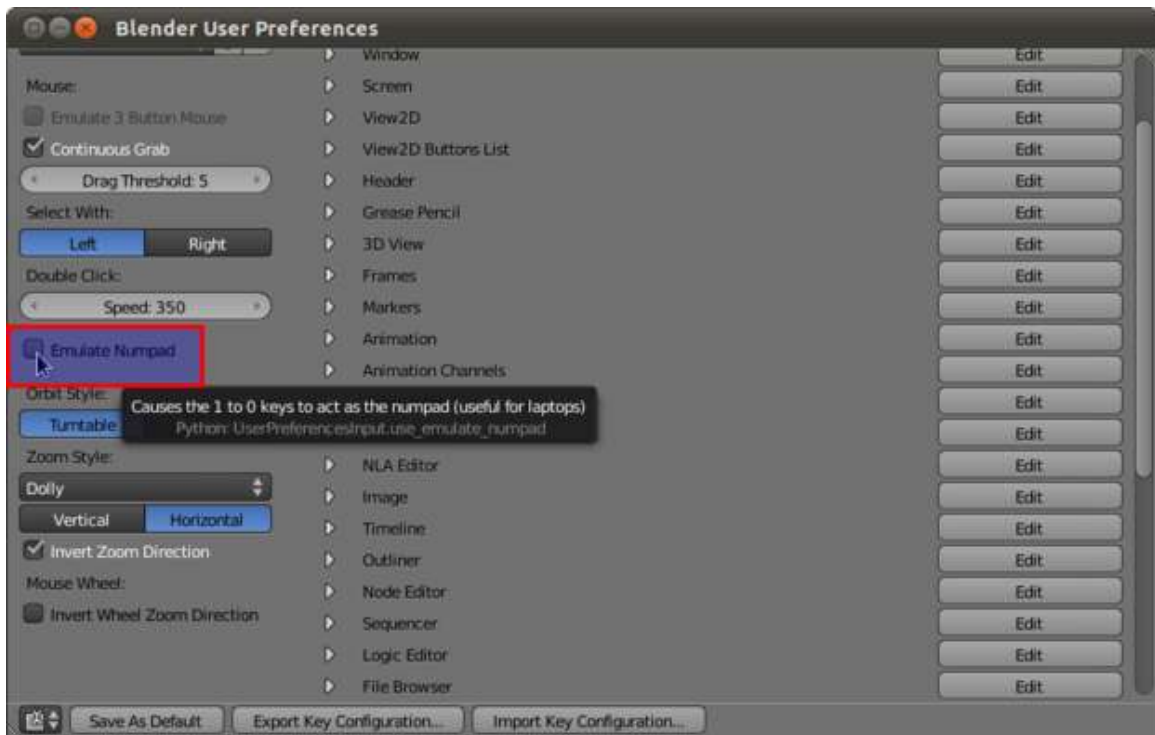


Figure 70

The numpad emulation option located under the *Input* tab allows you to use the keyboard number keys (the 0-9 keys above the qwerty keys) instead of the numpad keys. This feature is especially useful if you are using Blender from a laptop and do not have a standard numpad.

22. Properties Window (version 2.5)

1. REDIRECT Blender 3D: Noob to Pro/Buttons Windows¹

¹ Chapter 15 on page 75

23. 3D View (version 2.5)

The 3D View looks slightly different in Blender 2.5x:

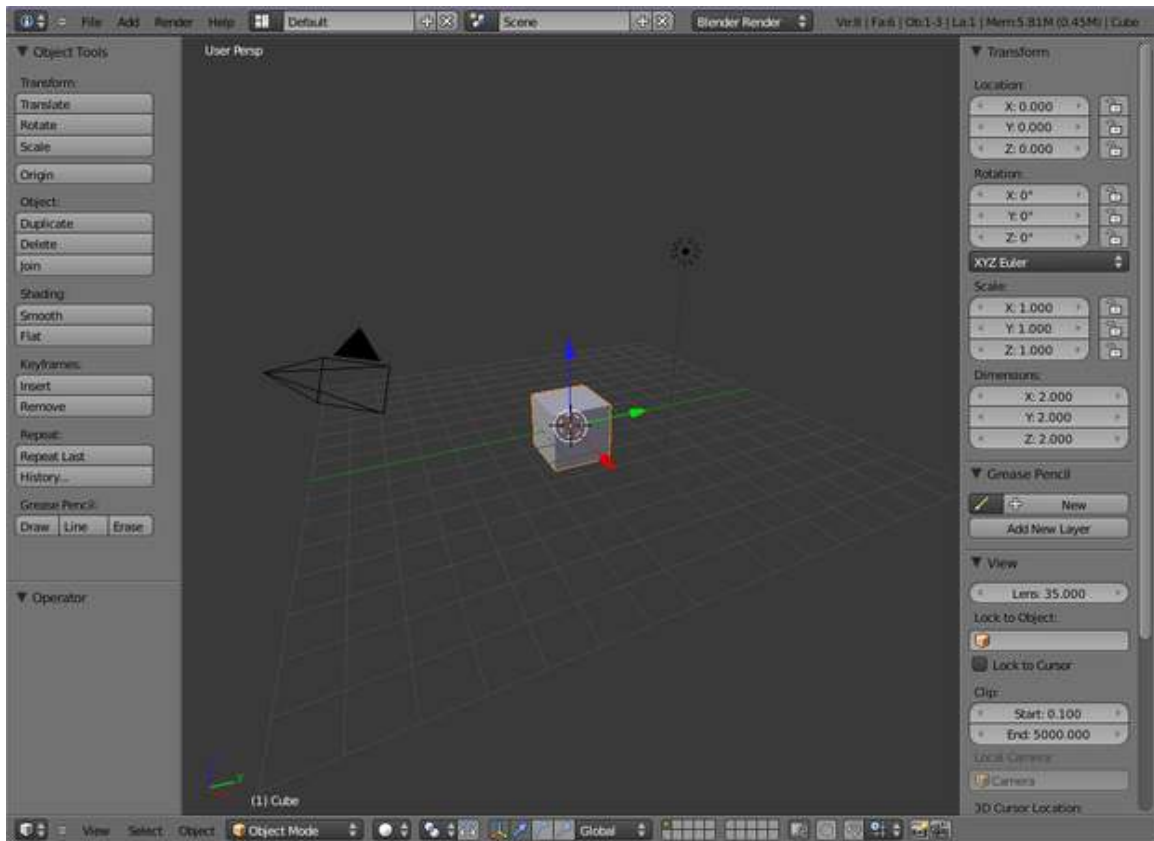


Figure 71

The obvious visible difference is the addition of *shelves*: the Tool Shelf on the left, and the Object Properties on the right. You can toggle the visibility of these by pressing **T** for the Tool Shelf and **N** for the Object Properties. Alternatively you can drag the inner edge of the shelf to the window edge to make it disappear. Whichever way you do it, when a shelf is hidden, this symbol

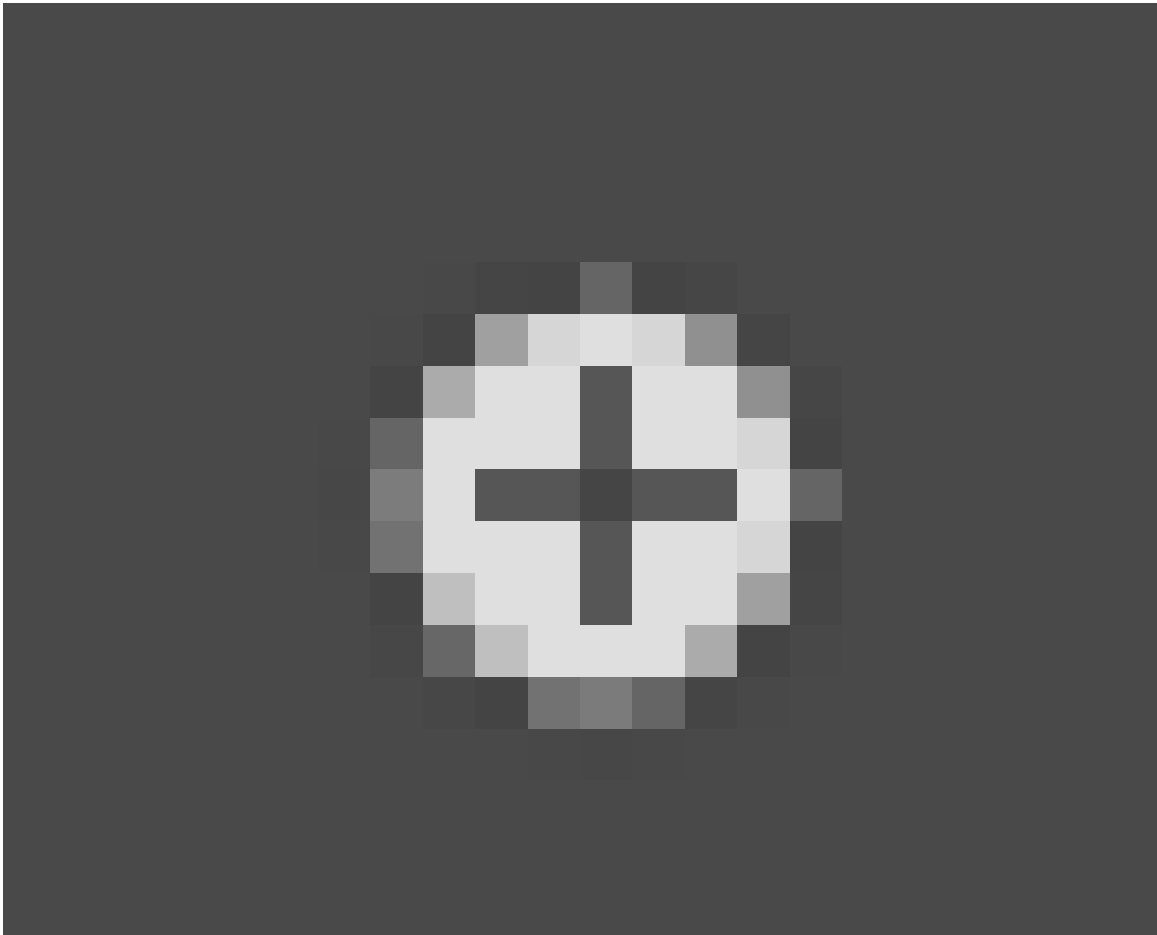


Figure 72



will appear in the corner of the window, and can be clicked to make the shelf reappear.

Note:

The Spacebar Menu: In previous versions of Blender, pressing the spacebar would bring up a copy of the main Blender menu hierarchy. In 2.5x, it brings up a scrolling, searchable menu of all applicable operators. Type some text into the box at the top to restrict the items shown to those containing the specified text as a substring.

Apart from the above, operating in the 3D view is still largely the same.

Note:

The Add menu is directly accessible via the  +  hotkey.

However, some keyboard shortcuts from older Blender versions may not work any more. They may yet make a reappearance as development of the 2.5x version continues. But in any case, most of the operations are now also accessible from the menus in the window header, so keyboard shortcuts are no longer the only way to get to them.

24. Blender Memory Management

24.1. Datablocks And Users

It is helpful to understand how Blender manages memory. Just about everything in a Blender document—objects in scenes, scenes themselves, materials, textures, whatever—is stored in a *datablock*. Each datablock may be referenced from one or more places, mostly in other datablocks—in Blender parlance, it has one or more *users*. For example, several different objects might share the same material, so when you change the characteristics of the material, it automatically changes the appearance of all those objects.

If the number of users of a datablock drops to zero, it still stays around in memory, but *it will not be saved when the document is saved*. Thus, if you save and reload the document, all the datablocks with zero users will disappear.

But up until that point, the datablock will continue to appear in the relevant popup menus, so you can reassign it to more users.

You can also assign a *fake user* to a datablock; this is what the “F” button is for in the popup menus that list datablocks of that type. This ensures that the user count never goes to zero, so the datablock always gets saved in the document even if it has no real users. This is useful for “library” documents, which can contain collections of useful materials and textures, say, that can be linked or imported into other documents, without also having to include dummy objects in the library just to ensure those materials and textures get saved.

For example, here is the widget that lets you choose the material for an object:

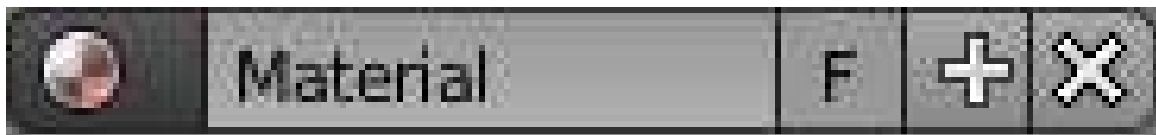


Figure 73

The main part shows the name of the current material, which is editable. The **X** button breaks the link to this material and decrements its number of users by one, while the **F** button assigns a fake user to this material. The **+** button lets you create a new material.

The material symbol on the left pops up a list of existing materials to choose from, plus a search box to search all existing materials:

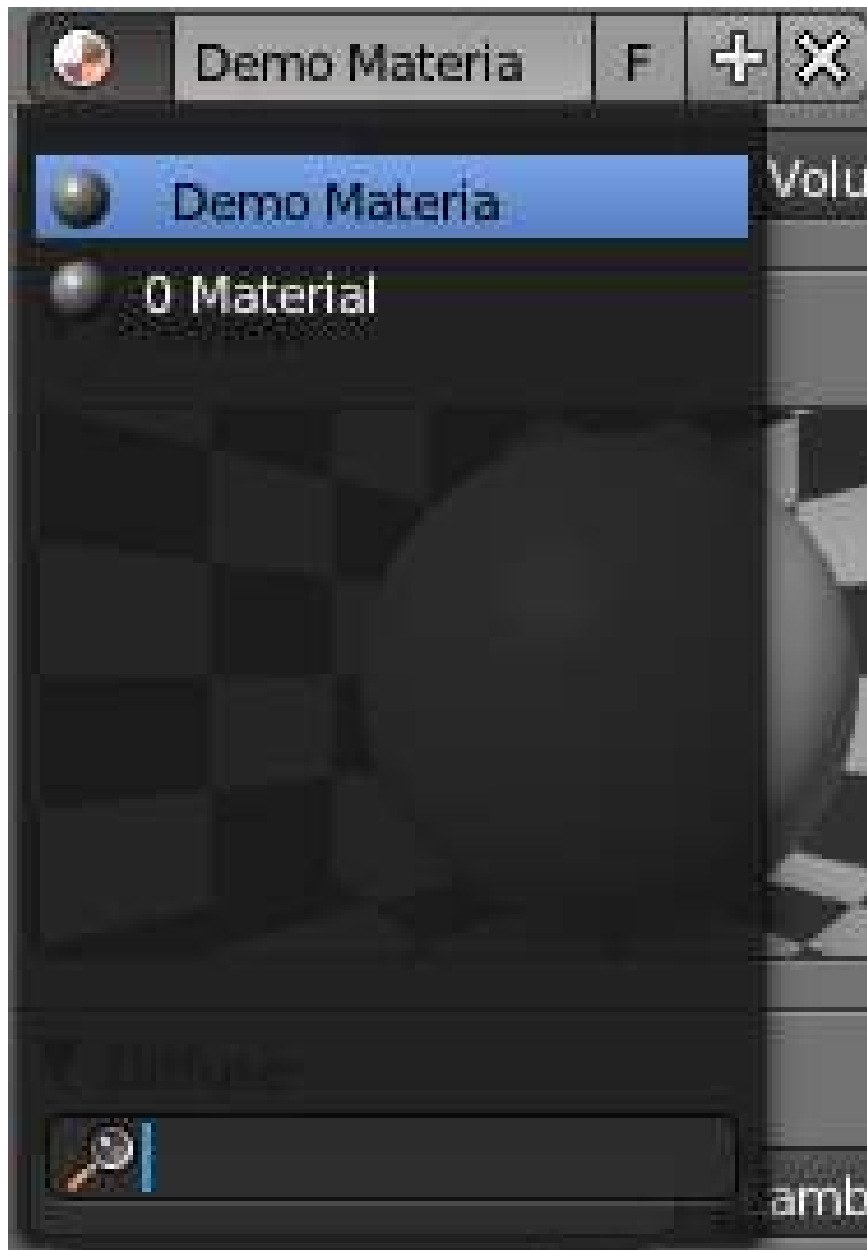


Figure 74

Note the entry with the **O** symbol next to it; that currently has a user count of zero, and will disappear when the document is saved and reloaded, if it is not further used.

The widget also displays the current user count if it is greater than 1:



Figure 75

In this case the count was incremented because the **F** button was selected.

This is the basis of the (slightly confusing) distinction in Blender between *object datablocks* and *object **data** datablocks*. Object datablocks contain the information common to all the *types* of objects in the 3D scene, regardless of whether they're mesh objects, lamp objects, camera objects or whatever; whereas the object *data* datablocks contain the information specific to that *instance* of the type of object, e.g. the vertex, edge and face definitions for this particular mesh you might be using, or the colour and energy of a lamp you've set up for your project, or the field of view of a camera you have.

Which leads us to the difference between the two object duplication commands, **SHIFT** + **D** and **ALT** + **D**: the former duplicates both the object datablocks and the object *data* datablocks (though this can be controlled in your User Preferences), while the latter only duplicates the object datablock. What that means is that in the first case the two objects are truly independent, but in the second case the new object continues to share the same object *data* datablock so a change in one will result in a change in both of them. So, for instance, if you use **ALT** + **D** on a mesh object and edit the vertices, edges or faces on one copy, the other copy will also be affected.

Part II.

Unit 2 - Basic Modeling and Shading

25. What is a Mesh?

The most fundamental step in the 3D development process is modeling, which entails creating 3D models of objects.

Blender supports many modeling techniques. "Mesh modeling" is the most basic and common modeling technique.

In this module, you'll learn the parts of a mesh, and you'll construct 2D meshes on paper and using Blender. You'll also learn how to create, select, and grab vertices in Vertex select mode.

25.0.1. Definitions

A **mesh** is a collection of vertices, edges, and faces that describe the shape of a 3D object:

- A **vertex** is a single point. (The plural of vertex is "vertices")
- An **edge** is a straight line segment connecting two vertices.
- A **face** is a flat surface enclosed by edges. (Some other applications call these "polygons")

25.0.2. An Exercise

To give you a feel for how the components of a mesh fit together, you'll now draw a 2D mesh on paper.

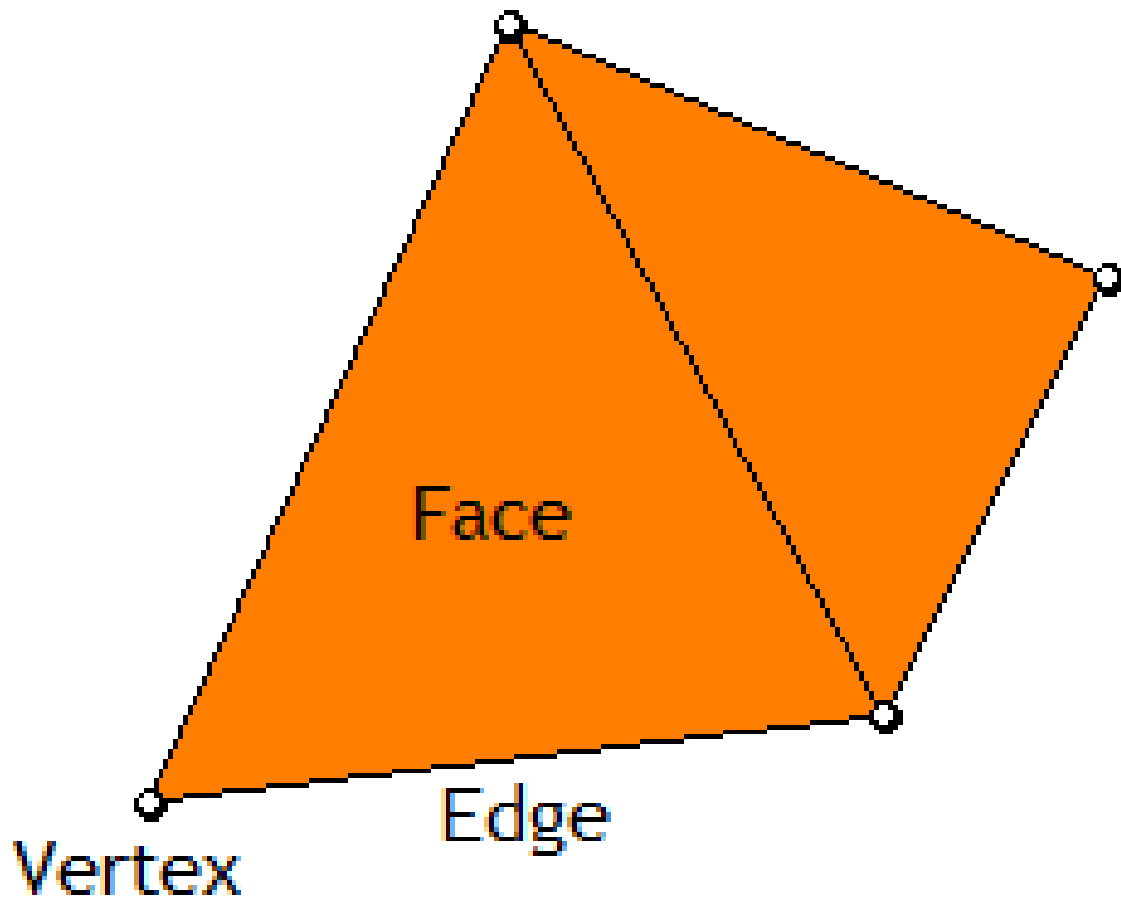


Figure 76 Your sketch might look like this.

1. Get a piece of paper and a pen or pencil.
2. Draw three dots that are a couple centimeters (about an inch) apart from each other.
 - Each dot represents a vertex in the mesh.
3. Connect two of the dots with a line segment.
 - The line segment represents an edge in the mesh.
4. Draw two more edges so that all three vertices are connected.
5. You've drawn a triangle; fill it in.
 - The area you filled in is a face.
6. Now draw the fourth vertex (dot) on the paper.
7. Connect the new vertex to two of the vertices you've drawn previously.
8. You now have another triangle; fill it in to create the second face.

Could you imagine creating a mesh of faces in 3D space? That's what mesh modeling boils down to.

You can keep filling up the paper with more vertices, edges, and faces if you want. You may want to try and create something interesting (like a letter of the alphabet) with your triangles.

25.0.3. More about Meshes

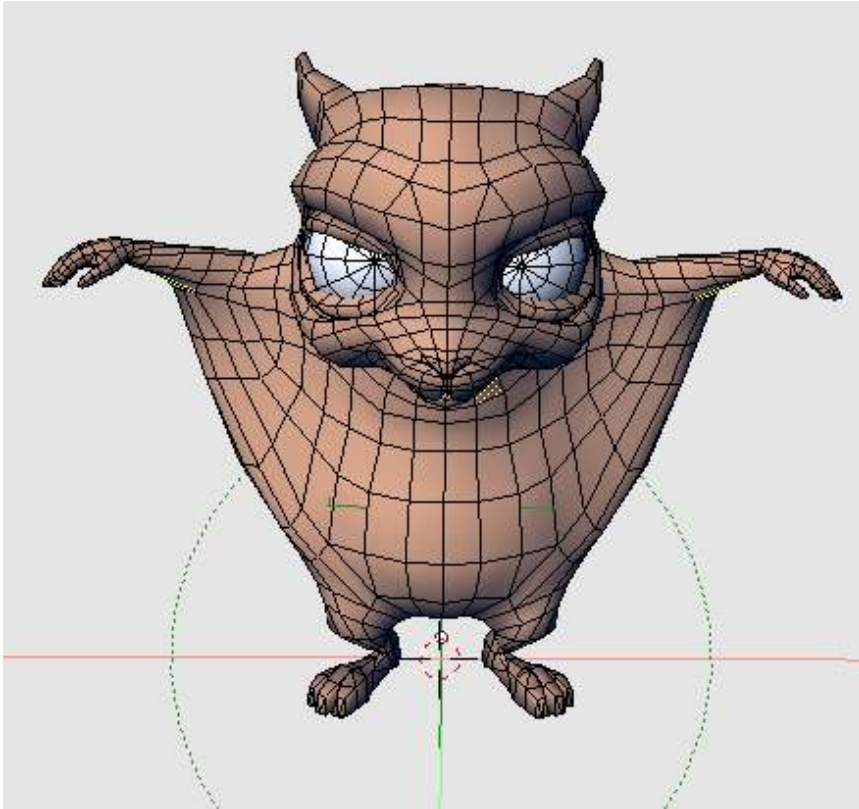


Figure 77 A character from the "Yo Frankie!" 3D video game in Edit Mode, showing edges and faces. The model is "(c) copyright Blender Foundation: apricot.blender.org".

Blender supports faces with three edges (triangles) or four edges (called **quads**). However, faces with five or more edges (so-called **N-gons**) are not supported. In order to model a hexagon with a Blender mesh, you would have to break it down into triangles and/or quads. (Latest versions of Blender are expected to support N-gons with "BMesh" feature. With the help of BMesh, to create a hexagon, you'll only need six vertices.)

Examine a 3D video game or CGI character for a while. Believe it or not, it is made up of little faces joined together. With modern technology, of course, there can be a *lot* of faces, so they may be tiny and hard-to-see. Surfaces that appear curved are composed of very many individual flat faces.

When you edit objects in Blender, you'll see every vertex and edge. However, vertices and edges are never rendered; only faces are rendered. The purpose of vertices is to provide 3D control points for faces.

25.0.4. Another Exercise

While it's possible to construct 3D meshes vertex-by-vertex, this is rarely done. However, doing it once the hard way will help you appreciate the powerful modeling tools built in by Blender. Along the way, you'll learn about Edit Mode and the grab tool, both of which you'll need in the next module.

First, summon the default cube:

1. Launch Blender.
2. If the NumLock indicator on your keyboard is unlit, press **[NUMLOCK]** so that numpad hotkeys will work properly.
3. Load the factory settings using *File* → *Load Factory Settings*.

(If you're unsure what **[LMB]** means, please review the "Keystroke, Button, and Menu Notation" module .)

Because you loaded the factory defaults, the 3D manipulator will be enabled. For mesh editing, it helps to turn the manipulator off:

1. Make sure the 3D View window is active.
2. Press **[CTRL] + [SPACE]** to toggle the manipulator on or off. You also could turn it on/off with the manipulator button on the 3D View header.

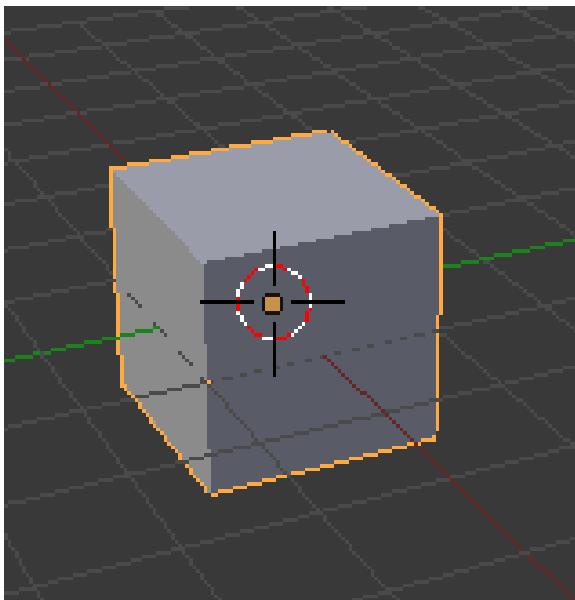


Figure 78 In Object Mode, the cube will look like this.

Because you just loaded the factory defaults, Blender should be in Object Mode with the default cube selected.

In order to modify the cube's mesh, you must put Blender into **Edit Mode**. (Edit Mode is a special mode for making changes to a single object.)

1. Press **[TAB]** once to go into Edit Mode on the cube.

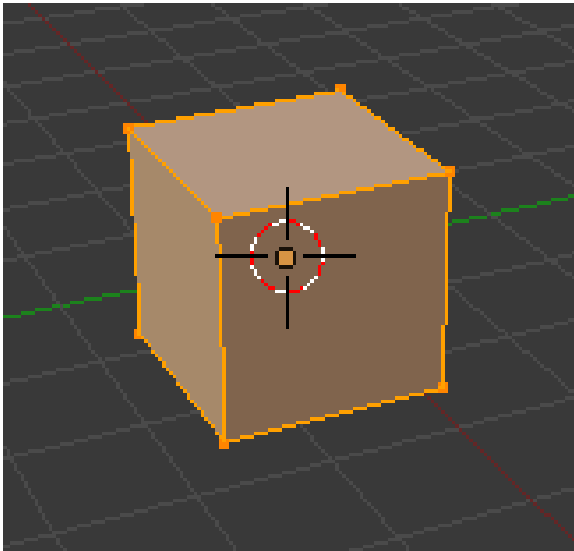


Figure 79 In Edit Mode, the cube will look like this.



Figure 80 Edit Mode indicator in the 3D View header.

Note:

`[TAB]` puts Blender into Edit Mode only if it's in a different mode to start with. If it's already *in* Edit Mode, `[TAB]` returns it to whatever mode it was in previously. So pressing `[TAB]` a second time would put Blender back into Object Mode.

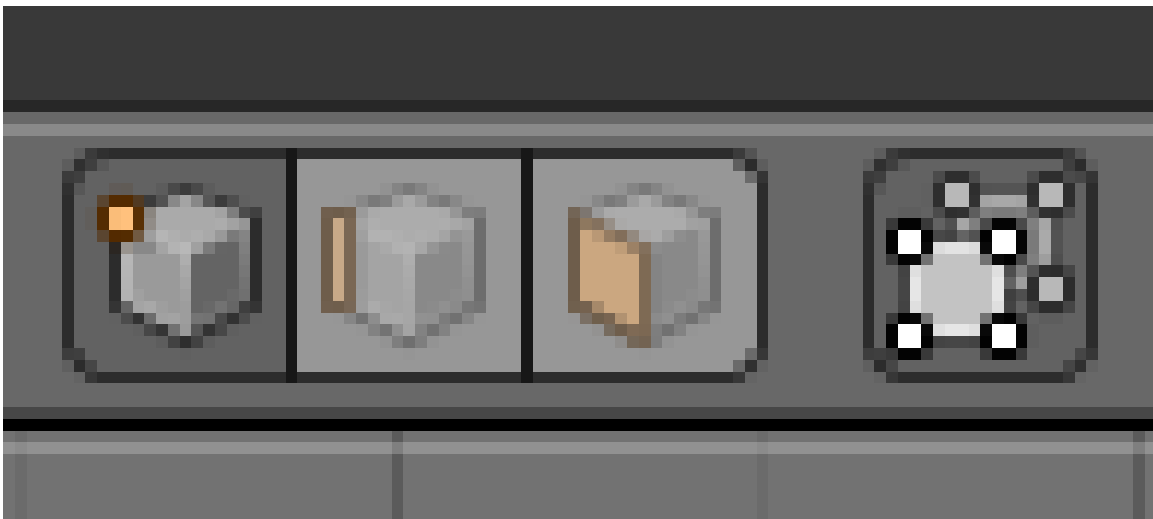


Figure 81 Select mode buttons in a 3D View header, showing Vertex select mode active.

Edit Mode has three (sub-)modes for selecting vertices, edges, and faces. Because you just loaded the factory defaults, you should be in Vertex select mode. In Vertex select mode, vertices show up as yellow, black, or white dots when they're selected and as pink dots when they're not. Because you just loaded the factory defaults, all eight vertices of the cube should be selected.

In Blender 2.59/2.60 the vertex(ices) that are either unselected/selected are the following colors: Unselected will be black; Currently selected will be white; Already selected (other than the current selection) will be orange. Also note that these colors correspond to edge selections as well.

To clear the boards for your first model, delete all of the cube's vertices:

1. Press **X**
2. A "Delete" menu will pop up. Choose *Vertices*.

Now you can repeat the previous exercise using mouse and monitor instead of pen and paper.

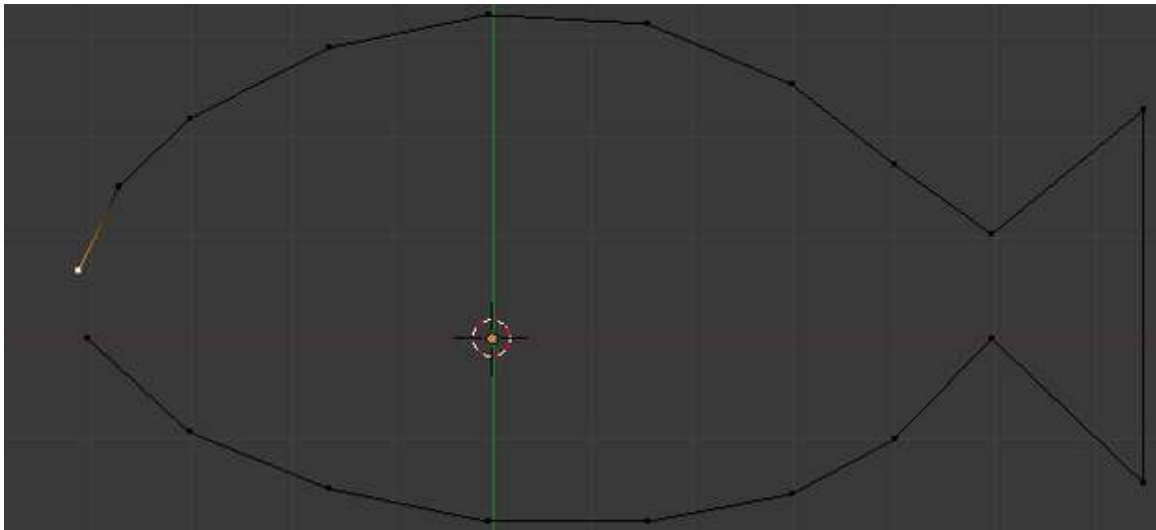


Figure 82 After step 3

1. Create a vertex by clicking with **CTRL** + **LMB**.
2. Create another vertex. Blender will automatically join the two vertices with an edge.
3. Create additional vertices (and edges) by clicking **CTRL** + **LMB**, but don't attempt to close a loop yet.

To create a face:

1. Press **A** to deselect all vertices.
2. Move the mouse to one of the vertices you want in the face.
3. Click **RMB** (or **CMD** + **LMB**) to select the vertex.
4. Move the mouse to another vertex you want in the face.
5. Click **SHIFT** + **RMB** (or **SHIFT** + **CMD** + **LMB**) to add it to the selection.
6. Continue adding vertices until you have three or four selected.
7. Press **F** to create the face.

Note:

Pressing **[A]** performs a “deselect all” operation *only* if something is selected. If *nothing* is selected, it performs a “select all” operation.

Note:

Pressing **[SHIFT] + [RMB]** does a "select" *only* if the vertex isn't already selected. If it is, it *deselects* the vertex.

Note:

If you make a mistake, you can undo your work step-by-step by pressing **[CTRL] + [Z]**.

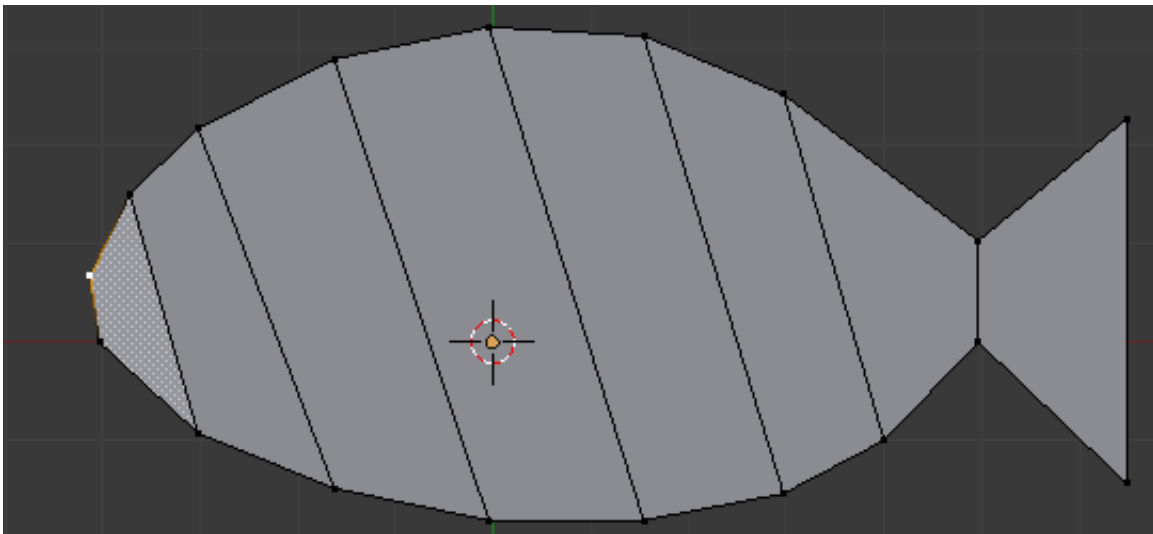


Figure 83 A completed 3D mesh

Create additional faces until all vertices belong to at least one face. Congratulations! You've just created your first 2D mesh in Blender.

You can reshape your mesh by moving vertices with the "grab tool".

To move one or more vertices:

1. Select the vertices using **[RMB]** and **[SHIFT] + [RMB]**.
2. Drag in the viewport with **[RMB]** (or press **[G]**) to activate the grab tool.

The 3D View header will be replaced by numbers: "Dx: 0.0000 Dy: 0.0000 Dz: 0.0000 (0.0000)". You're now using the "grab tool" and can drag the vertex around using the mouse.

The grab tool disables most of the normal hotkeys, so it's important to know how to get out. You can exit the tool at any time using:

- **[LMB]** or **[ENTER]** to confirm the changes
- or
- **[RMB]** or **[ESC]** to cancel the changes.

If you're up for a challenge, try making your mesh 3D:

1. Change the viewpoint, perhaps to one of the perfect views.
2. Add vertices and/or move the existing ones around in the new view plane.
3. When you're done, rotate to a new viewpoint to examine your work.

Now that you know how to create and grab vertices, you're ready for a quick lesson in extrusion and merging.

25.1. Additional Resources

w:Polygon mesh¹

¹ <http://en.wikipedia.org/wiki/Polygon%20mesh>

26. Quickie Model

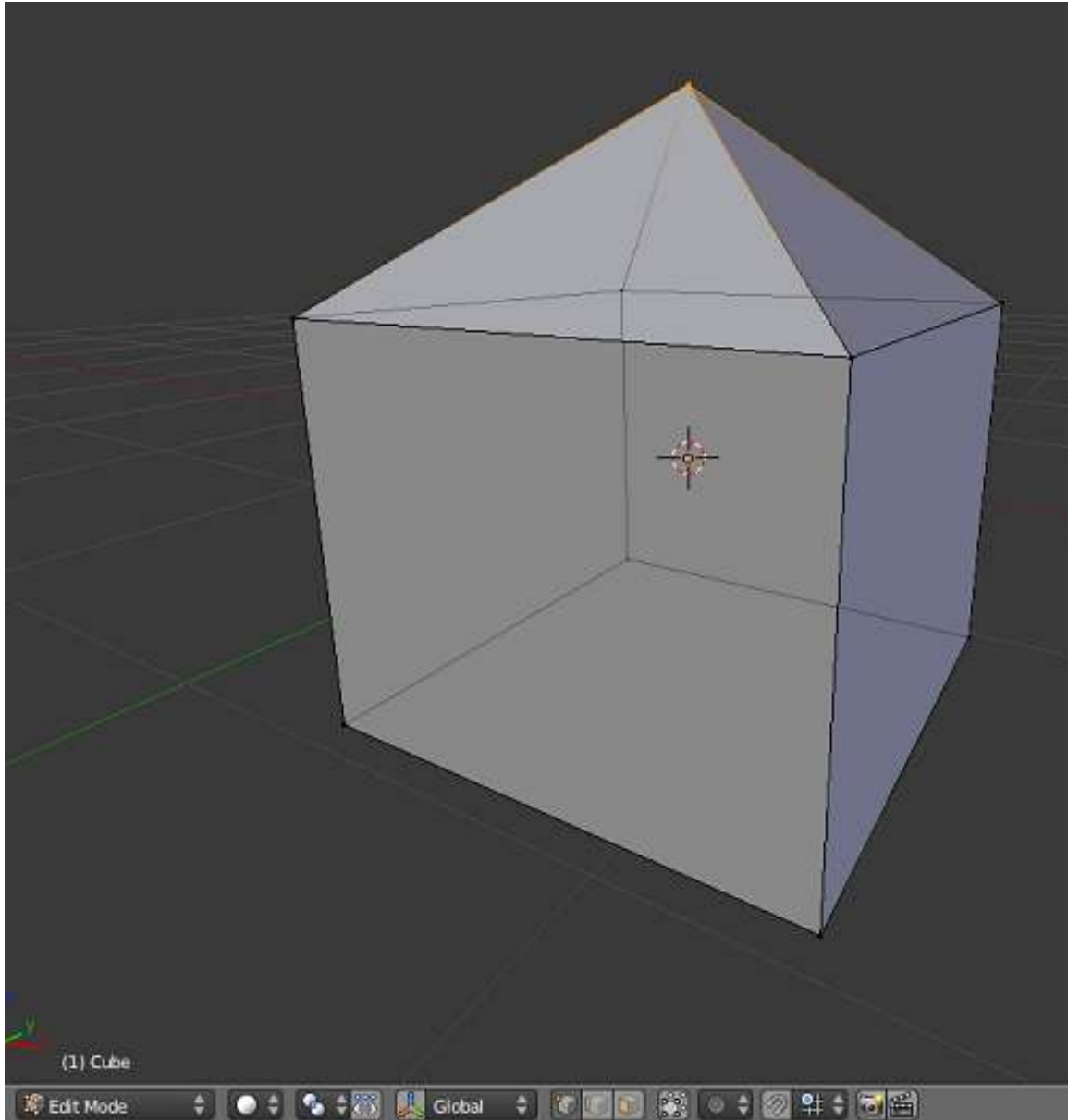


Figure 84 Your goal.

In this module, you'll learn how to extrude and merge vertices of a mesh and how to save a model. This module also introduces the File Browser window type.

Your first model will be a house, with four walls and a pyramidal roof. Simple! And because you're going to use the default cube as a base, all you actually need to build is the roof!

Editing in Blender generally involves four steps:

1. Selecting an object to edit.
2. Activating Edit Mode on that object.
3. Selecting part(s) of the object to act upon.
4. Specifying the action(s) to be performed on those parts.

26.1. Bring up the Default Cube

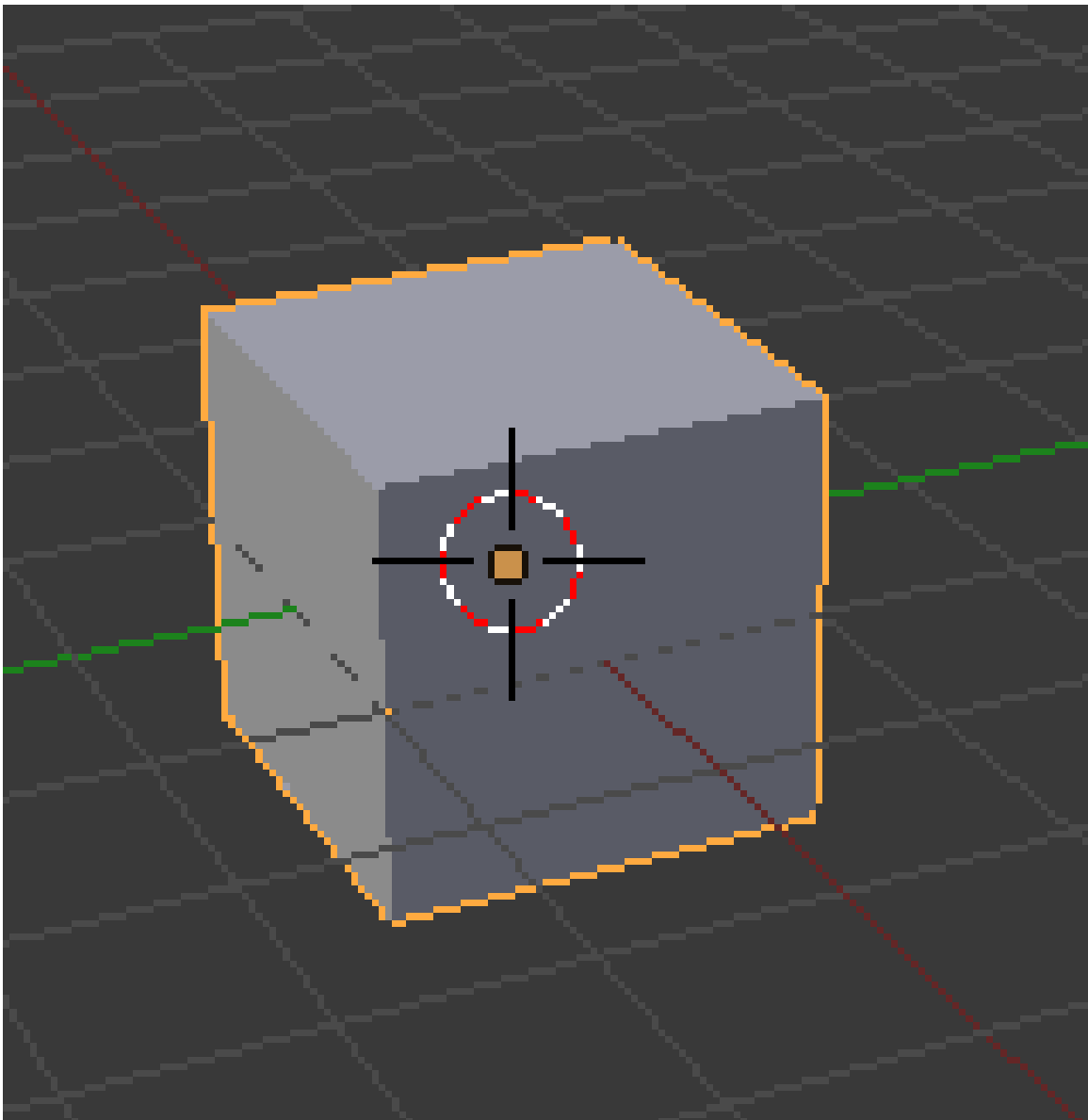


Figure 85 The default cube in Object Mode.

1. Launch Blender.
2. Load the factory settings using *File* → *Load Factory Settings*.

This should give you a perspective view of a scene containing three objects:

- a cube,
- a light source,
- a camera.

26.2. Setting up the Viewport

It will be easier to work on the roof of your house in a perspective side view:

1. Press **[NUM3]** to switch to a "perfect" right side view.

Note:

[NUM5] puts the viewport into perspective *only* if it's not *already* in perspective. Otherwise, **[NUM5]** switches the viewport *back* to orthographic view.

"Right Persp" will be shown on the top left of the 3D View. The "up" (*Z*) direction in the scene is now "up" on your monitor as well.

It will also help to zoom in a bit:

1. Make sure the 3D View window is active (which means your mouse cursor is in it).
2. **[SCROLL]** or press **[NUM+]** a few times until the cube is about 1/3 the height of the viewport.

Because you just loaded the factory defaults, the 3D transform manipulator will be enabled. For mesh editing, it will help to turn the manipulator off:

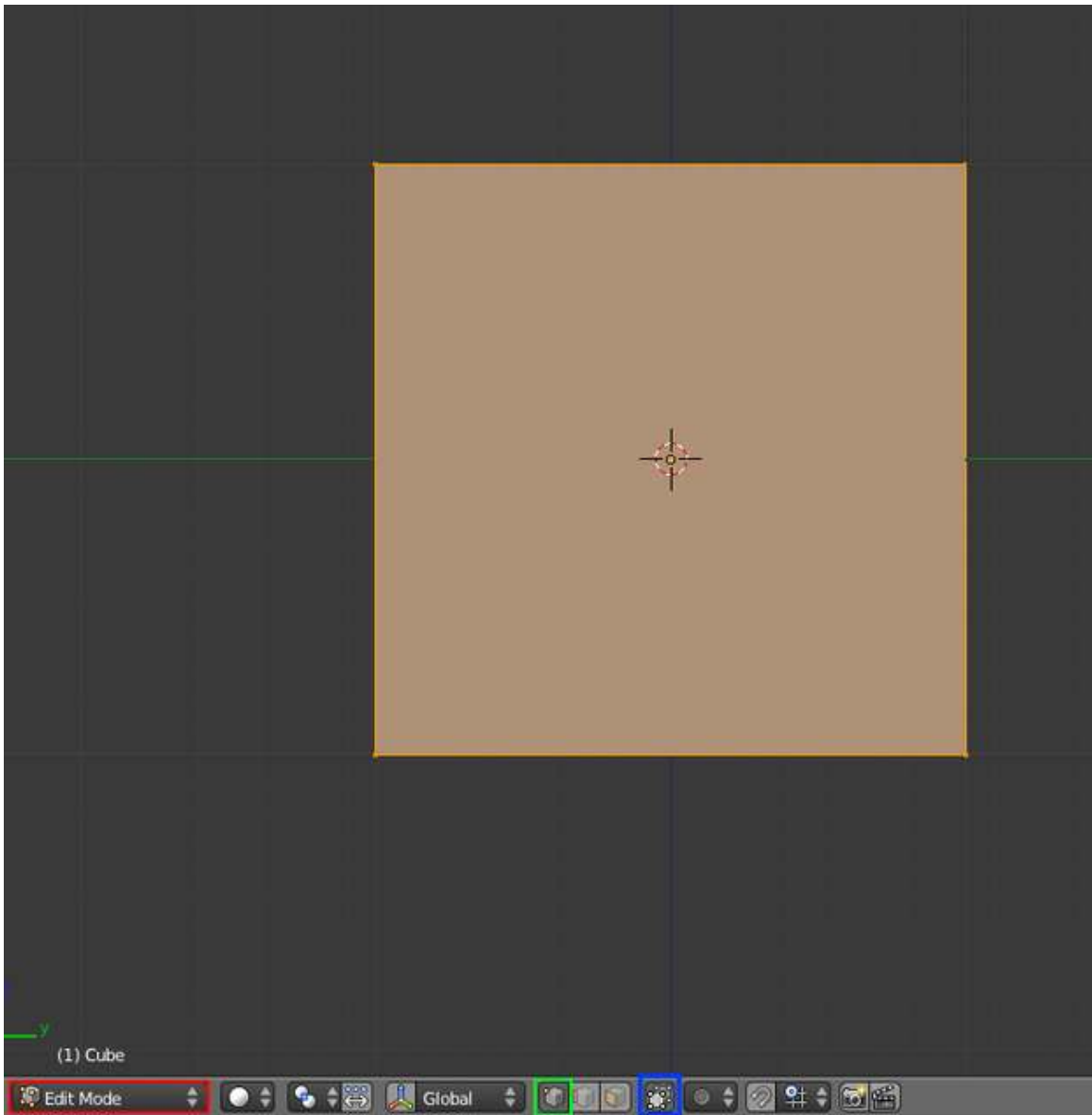
1. Make sure the 3D View window is active.
2. Press **[CTRL] + [SPACE]** to toggle the manipulator on or off.

Press **[TAB]** once. This puts you into Edit Mode on the selected object(s)—namely, the cube.

Note:

If the lamp and/or camera were selected instead of (or in addition to) the cube, you wouldn't be able to enter Edit Mode. (Cameras and lamps are edited in a different fashion.)

Here's how the cube should look at this point:

**Figure 86****Note:**

The *Oclude Background Geometry* button is only visible when Blender is in Edit Mode and the draw type is Solid, Shaded, or Textured.

The default cube is constructed as a mesh. Now that you're in Edit Mode, you can access the individual vertices, edges, and faces that make up the mesh. The default cube consists of eight vertices, twelve edges, and six faces.

26.3. Adjusting the Height

Right now, all eight vertices are selected, so any vertex edits you make will affect them all. For instance, if you were to move a vertex, the other seven vertices would follow. In order to build a roof peak for the house, you need to alter just the four top vertices of the cube. To do that, you must change the selection so that only *those* vertices are selected.

1. Turn 'Occlude Background Geometry' off, so you can see all vertices. Note, in newer versions, this button is called "Limit Selection to Visible."
2. Deselect the bottom four vertices, one by one, using **[SHIFT] + [RMB]**.

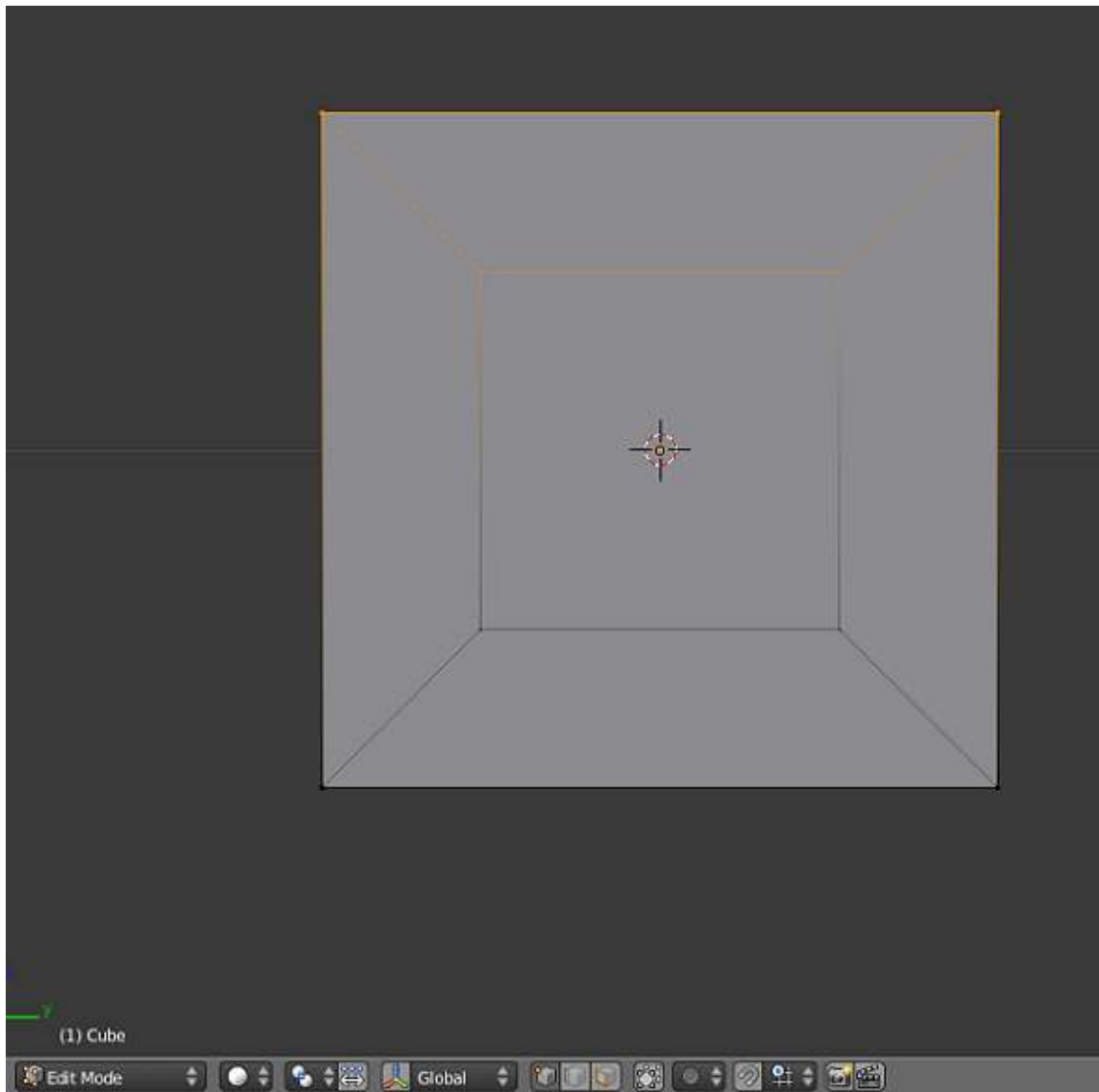


Figure 87 frame

The picture on the right shows the cube (in right perspective view and occlude background geometry "off") with the correct vertices selected.

Note:

Remember, if you make a mistake, you can undo your work step-by-step by pressing **CTRL** + **Z**.

Now you'll adjust the height of your house's ceiling. Activate the **grab tool**:

1. Make sure Blender is in Edit Mode, with the relevant part(s) of the object selected.
2. Make sure the 3D View window is active.
3. Press **G**.

The 3D View header will be replaced by numbers: "Dx: 0.0000 Dy: 0.0000 Dz: 0.0000 (0.0000)".

You want to lower the ceiling without making the walls crooked. This is hard to do freehand, but happily the grab tool provides an option for doing just that.

With the grab tool activated:

1. Press **Z** to limit movement to the global Z-axis.

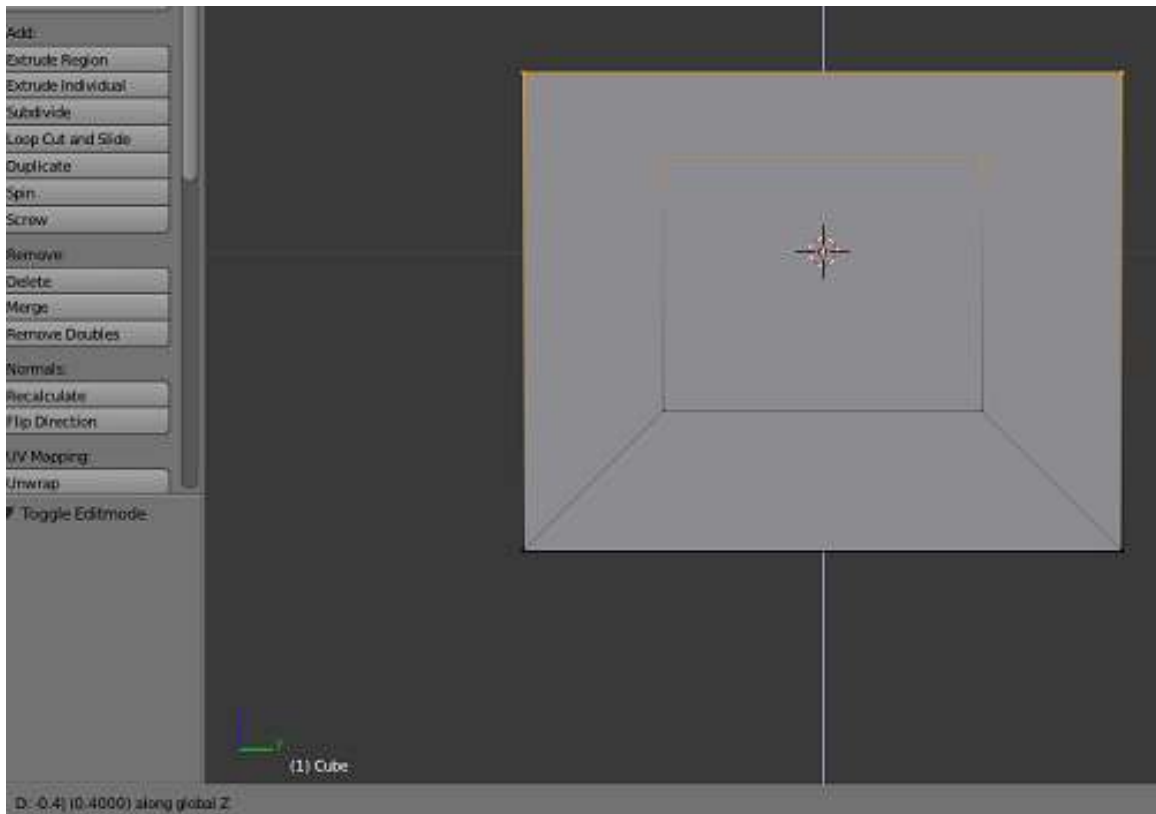


Figure 88

When your ceiling is the height you want, confirm the grab with **LMB** (or **ENTER**).

26.4. Extruding

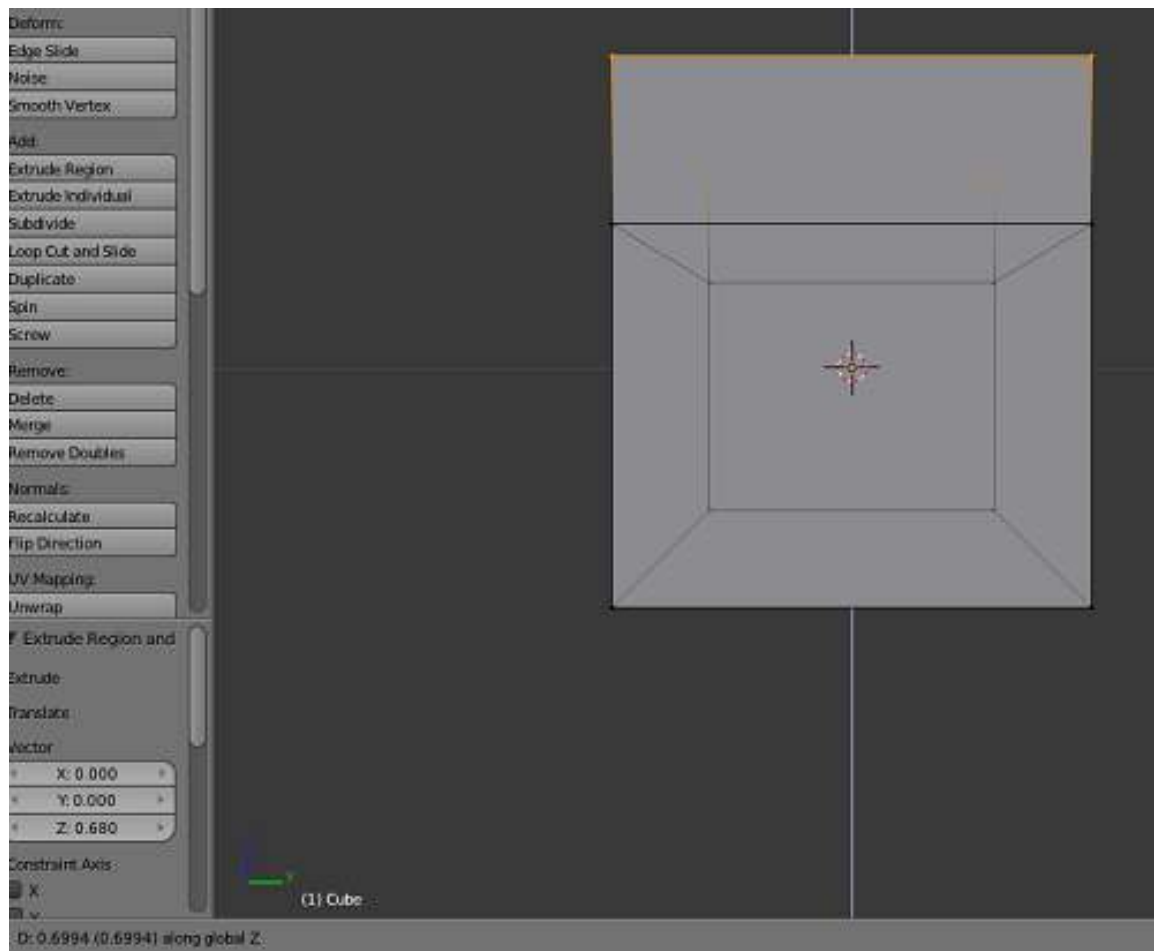


Figure 89 Step 7: the extruded attic, ready to confirm

Now you're going to "add on" to your house by **extruding**. Extrusion begins by duplicating selected parts of an object. Then the new parts are pulled away from the old ones, with new faces and edges created as necessary.

To add an attic to your house:

1. Make sure Blender is in Edit Mode, with the top four vertices selected.
2. Make sure the 3D View window is active.
3. Press **E** to activate the **extrude tool**.
4. Restrict movement to the Z axis and move the mouse pointer upward.
5. When the attic is the height you want, confirm the extrude with **LMB** or **ENTER**.

Note:

At the end of this process only the four new vertices (the upper corners of the attic) will be selected. The others (including the four that were originally selected) will not be selected.

Note:

If you cancel an extrude operation without confirming it, duplicate vertices and edges have already been created. If this isn't what you wanted, use **CTRL** + **Z** to undo the duplication.

26.5. Merging

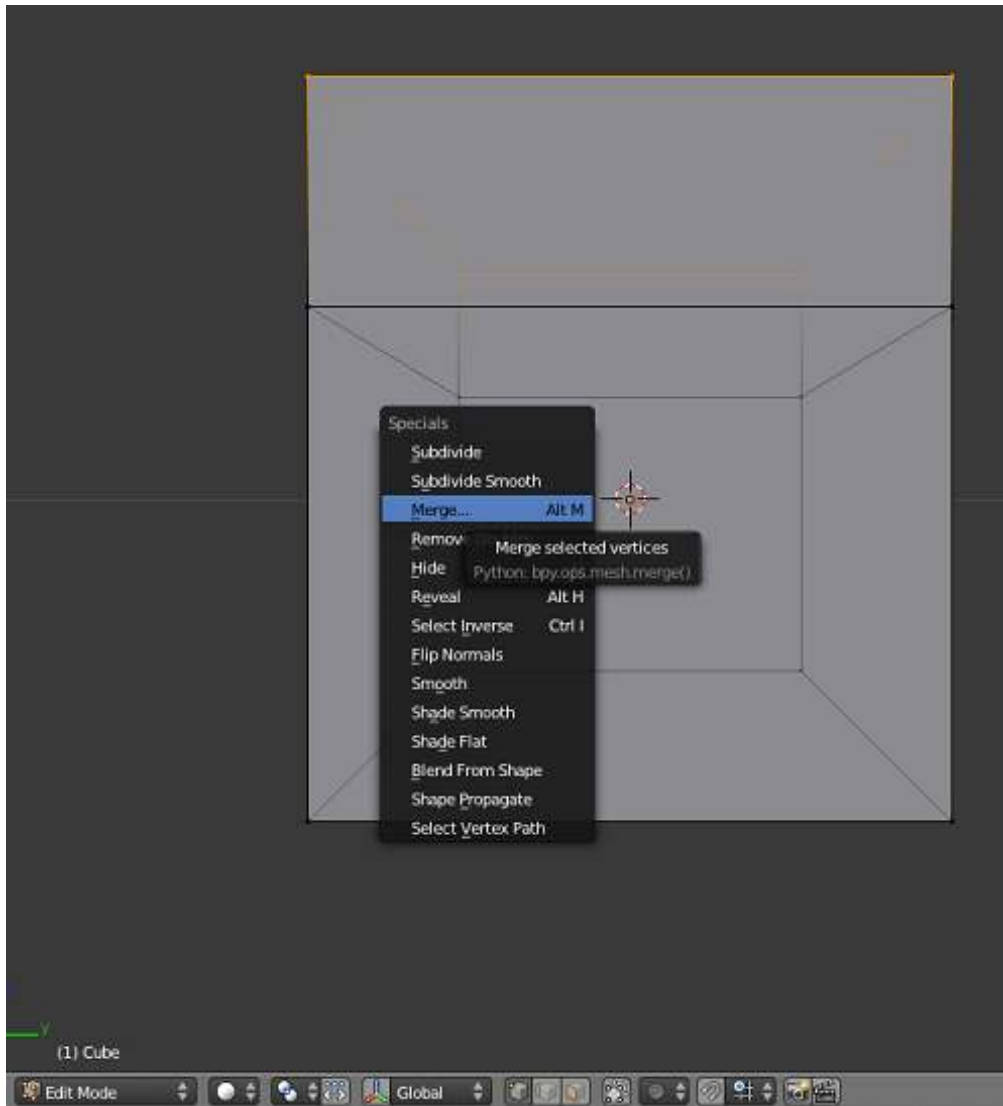


Figure 90 The Specials menu

You can change the roof from a flat one to a pyramidal one by merging the vertices of the roof:

1. Make sure Blender is in Edit Mode, with the four top-most vertices selected.
2. Make sure the 3D View window is active.
3. Press **W** to bring up the *Specials* menu.
4. Select *Merge*. (you can also access this by pressing **ALT** + **M**)

5. The *Merge* menu should pop up, select *At center*.

A message should appear on the Info header saying that 3 vertices have been deleted, this is because in order to merge four vertices into one, three vertices must be deleted.

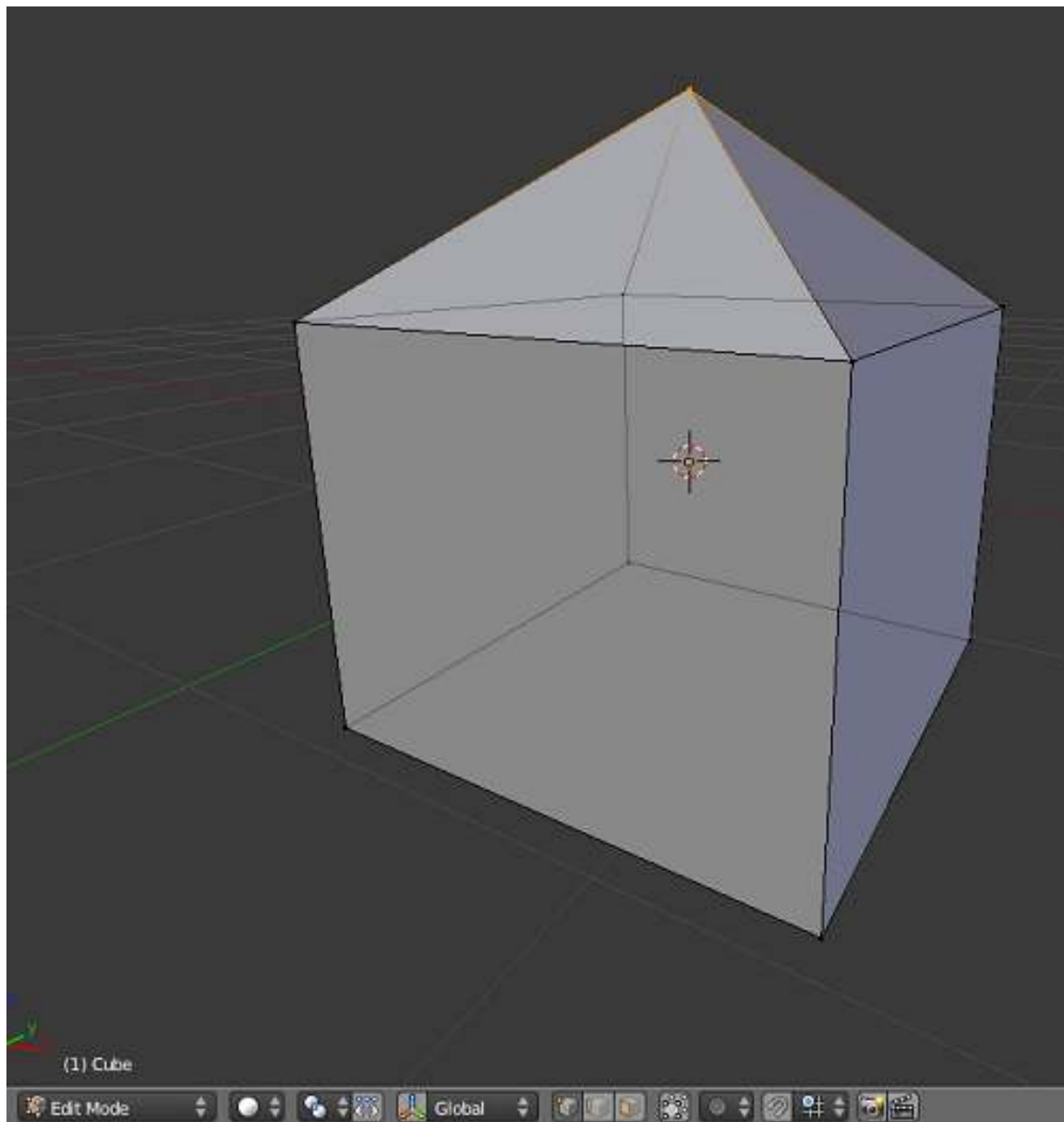


Figure 91 Your house now has a pyramidal roof!

27. Quickie Render

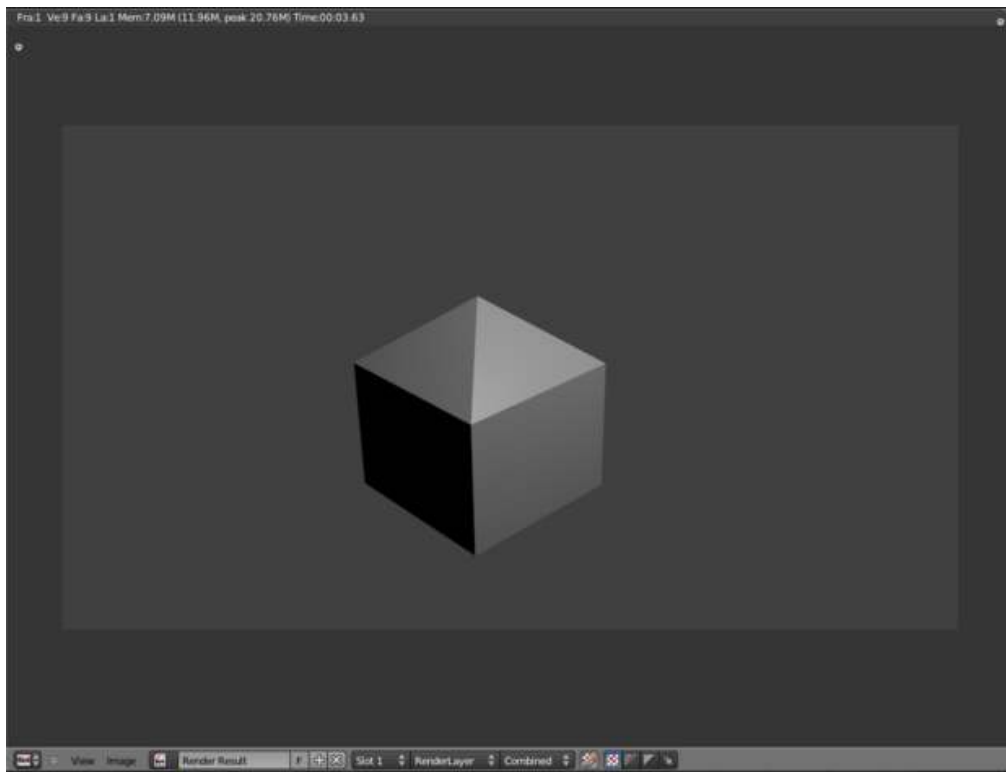


Figure 93

If you haven't completed the "Quickie Model" module, do so now. You will need the resulting model for this module.

Now that you've created your first model, you'll probably want to try rendering it. Your first render, with a single light source and only nine faces, should finish quickly. However, as your 3D scenes become more complex, you'll find that rendering can take a long time.

In this module, you'll render your quickie model and save the result in various file formats. You'll also learn how to aim cameras and create lamps.

27.1. Rendering the Quickie Model

1. Launch Blender and load factory settings.
2. To load the house model from the previous module:

- a) Press **F1** or select *File* → *Open*. This temporarily changes the active window into a File Browser window.
 - b) Navigate to the directory (folder) where you wrote the file, by clicking **LMB** on directory names in the File window. (Clicking on "." will take you up one level.)
 - c) Click **LMB** on the name of the file you wrote.
 - d) Click **LMB** on the "Open File" button. As soon as the operation is complete, the window will revert back to its former type.
3. Press **F12** or select *Render* → *Render Image*. This opens the **Image Editor** so you can watch the render progress.
- With the new Apple keyboard, use **FN** + **F12** to avoid the Mac Dashboard.
 - With Macintosh OS X 10.5, use **ALT** + **FN** + **F12**.
 - With Gnome, use **ALT** + **F12** to avoid the Gnome Search Dialog.

Note:

You can stop a render in progress by pressing **ESC** any time when the render window has the focus. Bear in mind that this will stop the render completely; you can't "resume" it by pressing **F12** again.

27.2. Aiming the Camera

If you don't get a picture of the house, or if the picture is not framed well, try moving or re-aiming the camera:

1. Press **ESC** to get back to Edit Mode, if needed.
2. Press **NUM0** to take the camera's viewpoint.
3. Press **SHIFT** + **F** to put the 3D View window into **camera fly mode**.

In camera fly mode, you can:

- **Pan** and **tilt** by moving the mouse pointer up, down, left, or right.
- Accelerate by **SCROLL** forwards.
- Decelerate by **SCROLL** backwards.
- Press any key or button to exit fly mode.

When you're done positioning the camera, try rendering again.

27.3. Lighting

If your cube is completely black, you may not have a lamp in the scene. Either the default lamp got deleted, or you're using a version of Blender that doesn't provide a default lamp.

To add a lamp:

1. Make sure Blender is in Object Mode.
2. Place the 3D cursor where you want the lamp to go; or add the lamp then immediately grab it, and move it somewhere else.

3. Press **SHIFT** + **A** .
4. In the popup menu, select *Lamp* → *Point*.

27.4. Saving the Render

Saving the scene (with **F2** , for instance) does not save any renders. Saving renders is a separate step.

To save your current render:

1. Make sure you are in the Image Editor
2. Press **F3** . This temporarily changes the active window into a File Browser window.
3. Navigate to the directory (folder) where you want to write the file.
4. Type a filename in the text box (to the left of the "Cancel" button).
5. To the left of the window, choose your preferred file type.
6. Click **LMB** on the "Save as Image" button. As soon as the save operation is complete, the window will revert back to its former type.

27.5. Image File Formats

JPEG¹ is a lossy² image format. For best image quality, you'll want to use a lossless³ format such as PNG⁴.

To change the render file format:

1. Click the camera icon on the top of the Properties panel to display the render options.
2. Find the *Output* dropdown menu.
3. Click **LMB** on the popout menu with the current file format (probably JPEG).
4. Select your preferred format. If you're doing an animation, then select one of the Movie codecs (such as AVI or MPEG).

27.6. Additional Resources

- the "Output Formats" module
- Tutorial on Using Multiple Cameras⁵ ← Pictures are missing from this tutorial
- Ira Krakow's Basic Blender Camera Positioning (Rigging)⁶

1 <http://en.wikipedia.org/wiki/JPEG>

2 <http://en.wikipedia.org/wiki/lossy%20compression>

3 <http://en.wikipedia.org/wiki/lossless%20data%20compression>

4 <http://en.wikipedia.org/wiki/Portable%20Network%20Graphics>

5 <http://forum.nystic.com/viewtopic.php?f=34&t=2966#>

6 <http://www.youtube.com/watch?v=NzEy2sCzcHo>

28. Understanding the Camera

Blender's camera controls the view of your model that appears in the render. If you have experience with photography, most of the quirks of real-world cameras do not apply in Blender (which is fortunate in some ways, unfortunate in others). But one important characteristic does still apply, and that is *field of view*.

There are two ways to control how much of the scene or model appears in the picture: either move the camera closer to it or further way; or change the *angle* of the lens. You do the latter in the Camera panel, which appears under the Editing buttons when a camera is selected.

One fact worth noticing is that, good camera handling and positioning is vital to the overall render of your work. Excellent camera skills can result in outstanding images and animations. Knowing where to put your camera at what time and at what interval will greatly make you a good artist and an animation expert.

You can learn this art from our natural environment. 3D animations are a good start to help you understand some of the basic camera positioning. Its a pretty nice art, and you've got to develop it, whether you're talented or not in that!

Remember, too, that too much of everything can be bad. The fact that you can move your camera to and fro without paying a penny does not grant you the permission to place your camera at wrong distances from your object of focus. Moving too far away from the object under consideration may also make it appear far too small. Position your camera well. To some extent, in Blender, WYSIWYG - What You See Is What You Get! I can bet on that. Trust me!

Here are two renders of the same scene with two different cameras, to illustrate the difference.

This one moves the camera closer but gives it a wider field of view:

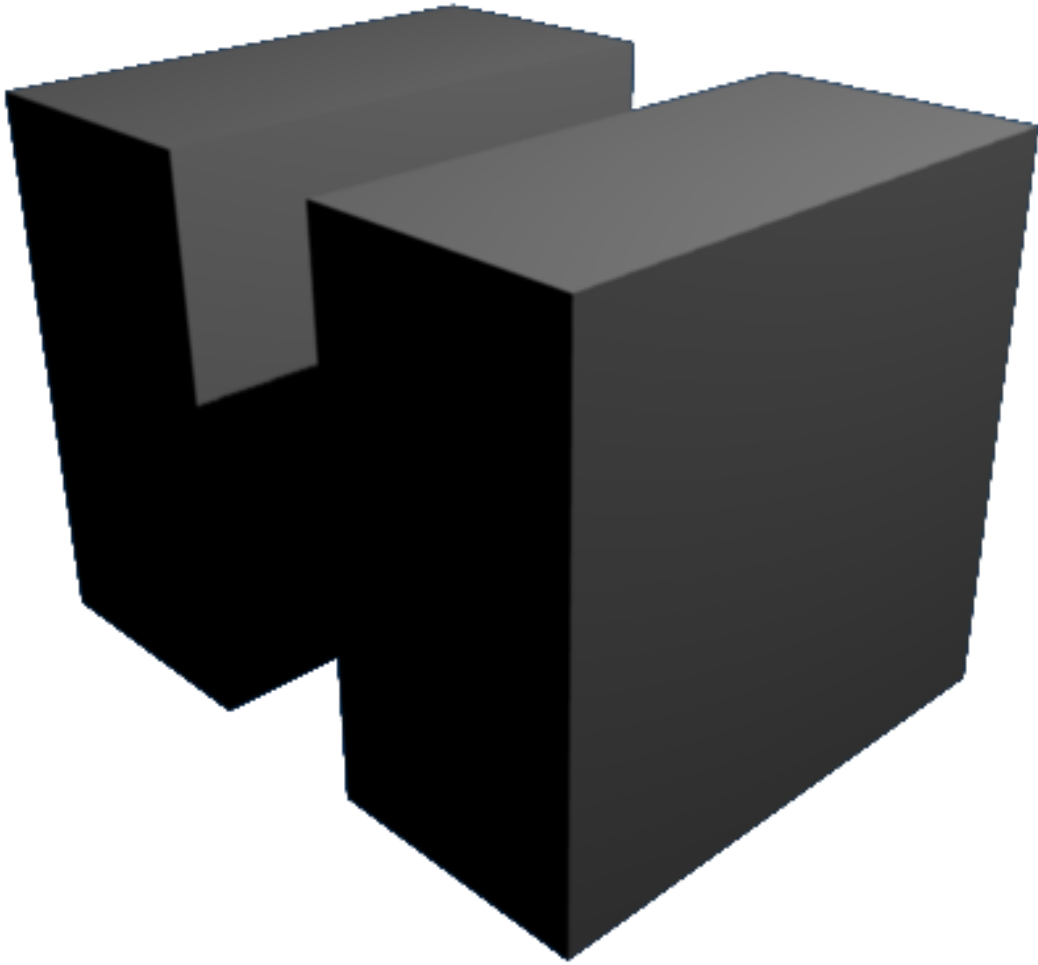


Figure 94

This one moves the camera back, while narrowing its field of view.

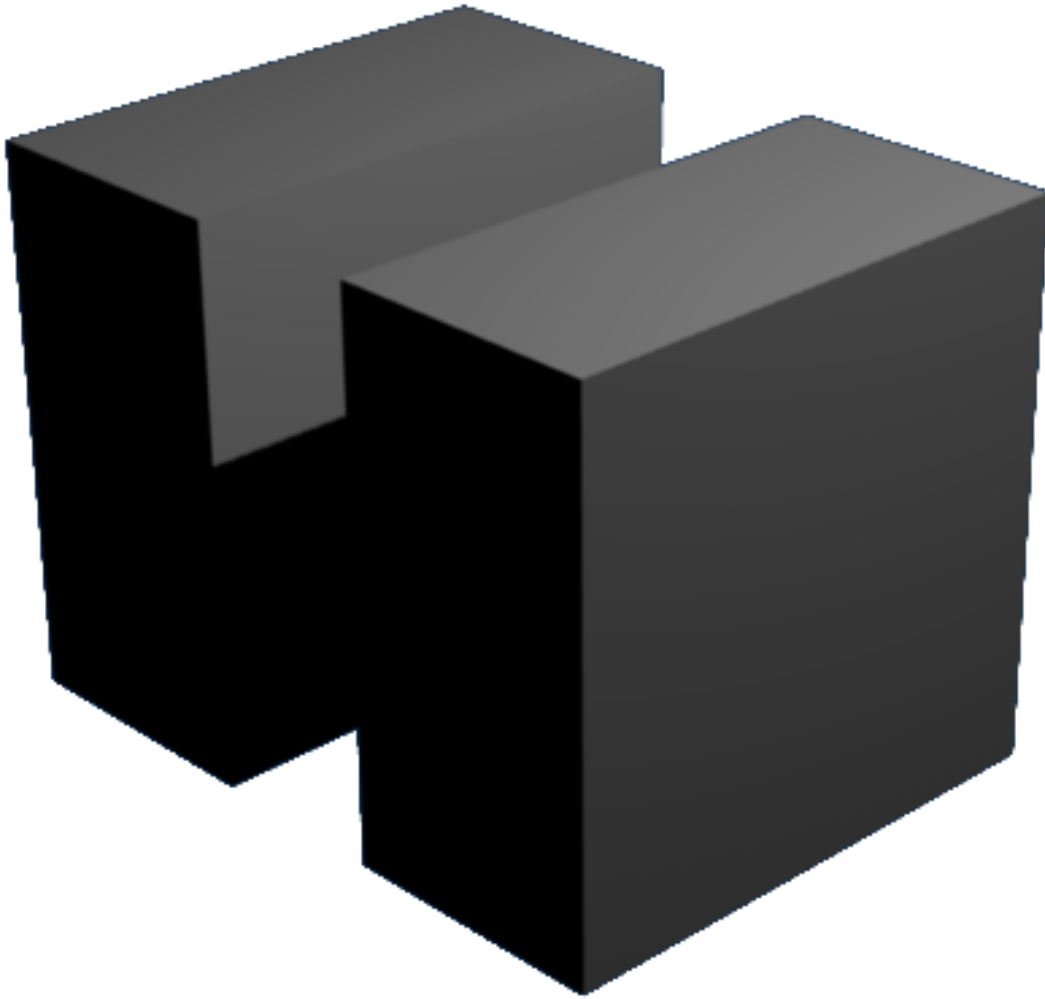


Figure 95

The latter is like using a “telephoto” lens with a real camera. Notice how moving the camera closer gives you a greater perspective effect.

es:Entendiendo la cámara¹

¹ <http://es.wikibooks.org/wiki/Entendiendo%20la%20c%Elmara>

29. Improving Your House

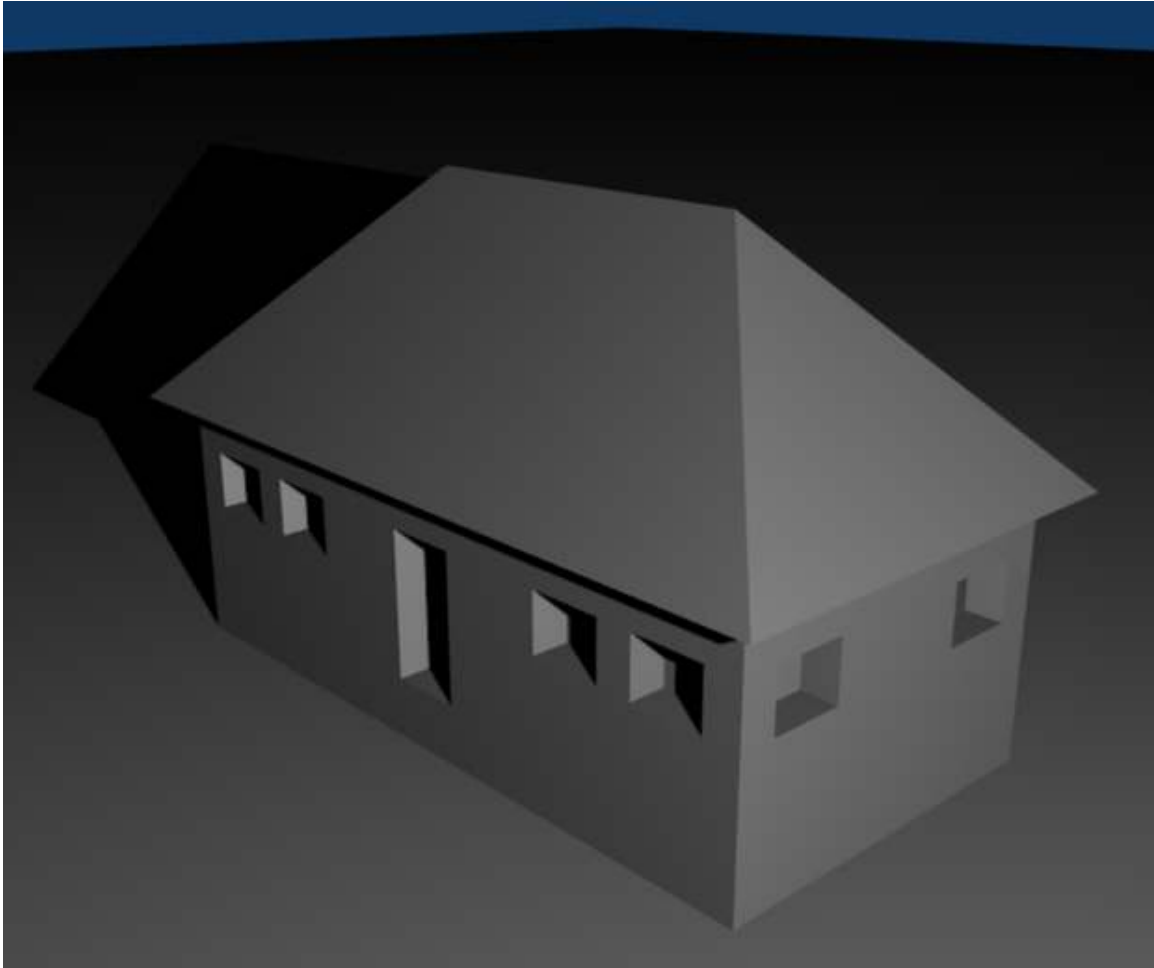


Figure 96 Your goal.

In this module, you'll refine the house model you created two modules ago. In the process, you'll learn how to access Blender's predefined meshes and how to set a pivot point. You'll also learn how to select, extrude, delete, and subdivide the edges and faces of a mesh model.

To begin, set up Blender as follows:

1. Launch Blender and load factory settings.
2. If you have a numpad, make sure NumLock is on.
3. Load the house model you created in the "Quickie Model" module .
4. If the 3D manipulator is active, disable it.
5. Adjust the viewpoint until you can clearly see two walls of the house and two sides of the roof.

29.1. Adding a Ground Plane

Your house needs some ground to rest on. You can model the ground as an object in your scene. Blender has many predefined mesh objects built in. Happily, one of these is a flat, square surface.

Recall that new objects are added at the 3D cursor. Before creating the ground, you should position the cursor at ground level:

1. Select the house by clicking **RMB** on it.
2. Enter Edit Mode by pressing **TAB**.
3. Select one of the bottom vertices by clicking **RMB** on it.
4. Bring up the *Snap* menu by pressing **SHIFT** + **S**.
5. Choose *Cursor to Selected*.
6. Leave Edit Mode by pressing **TAB** so that your ground is created as a separate object.

Note:

There's an "Aligned To View" setting in the "Editing" tab of User Preferences which is off by default in 2.5x. When this setting is on, the orientation of new objects depends on the current viewpoint. Pre-2.48a releases of Blender had "Aligned To View" on by default. If you've turned this setting on (or are using an old release) go to "top view" (by pressing **NUM7**) before creating the ground object.

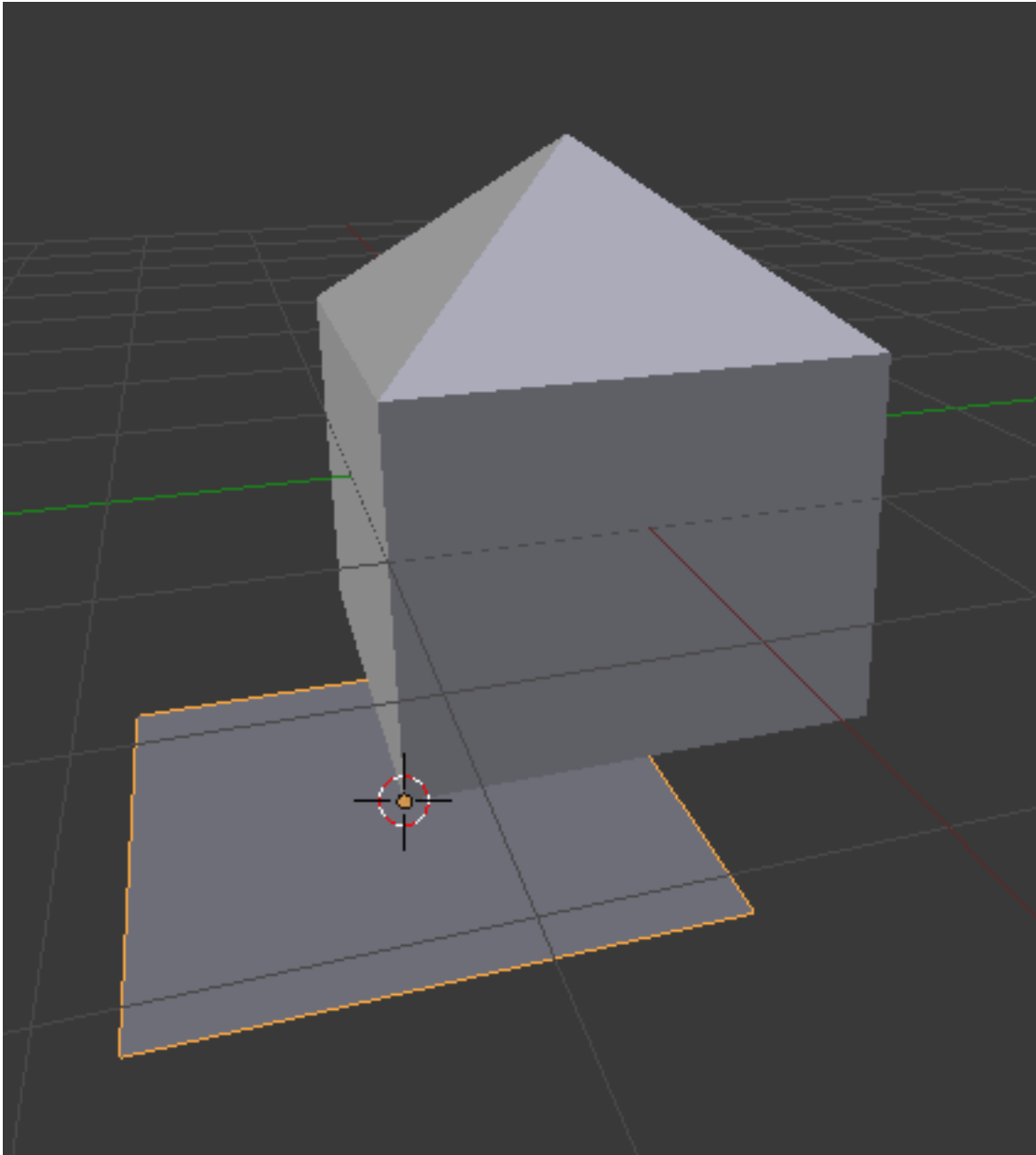


Figure 97 The ground added.

Now create the ground object:

1. Activate a 3D View window.
2. Press **SHIFT** + **A**.
3. Choose *Mesh* → *Plane*.

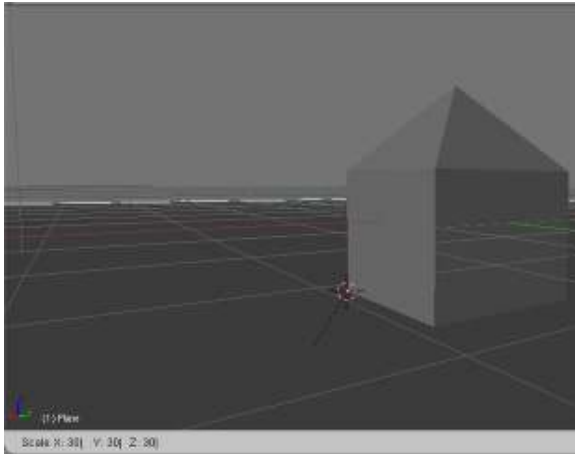


Figure 98 The scaled ground.

To enlarge (or scale) the ground object, use the **scale tool**:

1. Make sure Blender is in Object Mode.
2. Select the ground by clicking **[RMB]** on it.
3. Activate the scale tool by pressing **[S]**.
4. Type **[7KEY]** to enlarge the ground 7x.
5. Press **[ENTER]** or **[LMB]** to confirm and exit the scale tool.

29.2. Scaling with a Pivot

Suppose you want to shrink the house by 50%. As you can probably guess, this would be done with the scaling tool. However, if you did so right now without the right pivot point, the reduced house would no longer rest on the ground. Blender scales (and rotates) objects around a **pivot point**, which by default is located at the **median point** (geometric center) of the selected object(s).

In order to scale the house while keeping its base on the ground, you need the pivot point to be at ground level. Since the 3D cursor is at ground level, you can do this as follows:



Figure 99 Origin to 3D Cursor Menu Item.

1. Make sure Blender is in Object Mode.
2. Select the house by clicking **[RMB]** on it.
3. In a 3D View header, click **[LMB]** on menu item "Object" and put the mouse cursor over *Transform* and select *Origin to 3D Cursor* from the pop-up menu. This can also be done with **Ctrl+Shift+Alt+C**, select *Origin to 3D Cursor* from the pop-up menu.

The origin of the house is now at the center of the 3D cursor. If you scale the house, the place where the 3D cursor is located will remain fixed, and everything else will expand or contract from that point.

29.3. Edge Selection

It is often useful to select edges instead of vertices.

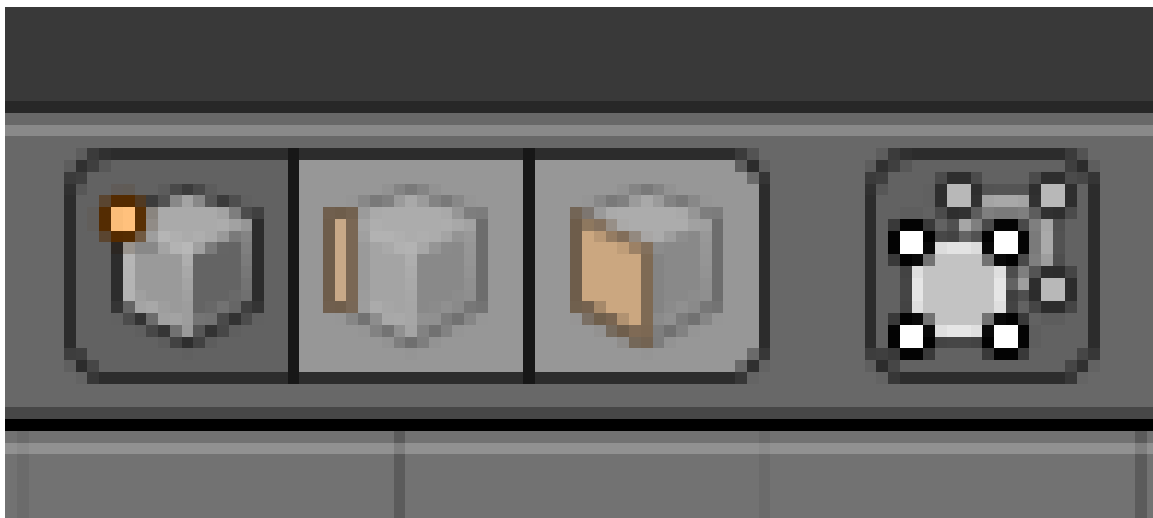


Figure 100 The select mode buttons.

1. Make sure Blender is in Object Mode.
2. Select the house by clicking **[RMB]** on it.
3. Press **[TAB]** to enter edit mode.
4. Click **[LMB]** on the Edge select mode button in the 3D View header.

In Edge select mode, edges appear as orange or white line segments when they're selected and as black line segments when they're not.

Just as you selected vertices in Vertex select mode, you can now select (and deselect) edges in the same way as vertices. This is also the same for Face select mode.

It may be because those edges are **doubled**. This can happen if you cancel an extrude operation and forget to undo the duplication. Here's a solution:

1. Switch to Vertex select mode.
2. Activate a 3D View window.
3. Select all vertices by pressing **[A]** once or twice.
4. Press **[W]** to bring up the "Specials" menu.
5. Choose *Remove Doubles*.

29.4. Extruding Edges

You can extrude edges in much the same way as you extrude vertices.

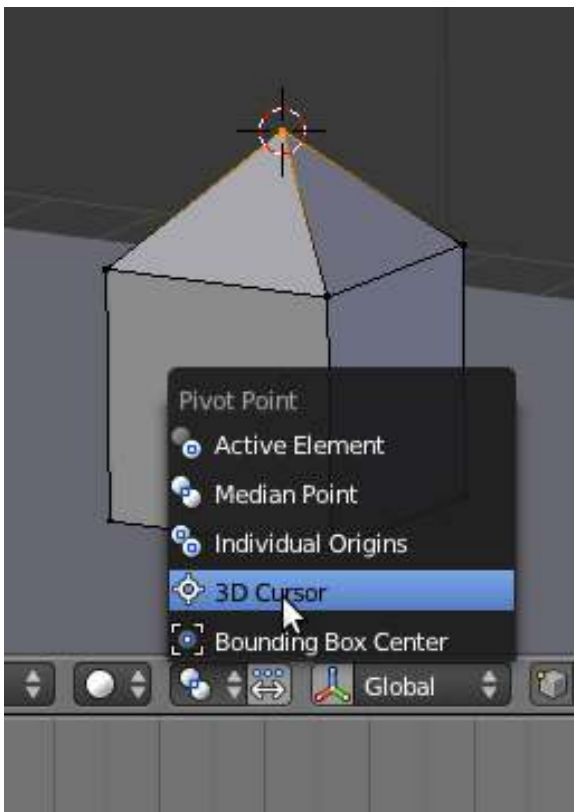


Figure 101 Step 5

To add an overhang to the roof of your house, first move the pivot point to the peak of the roof:

1. Switch to Vertex select mode.
2. Select just the vertex at the peak of the roof.
3. Press **SHIFT** + **S** to bring up the Snap menu.
4. In the Snap menu, choose *Cursor to Selected* to move the 3D cursor to the peak.
5. Use the "Pivot" menu in the 3D View header to change the pivot to "3D Cursor".

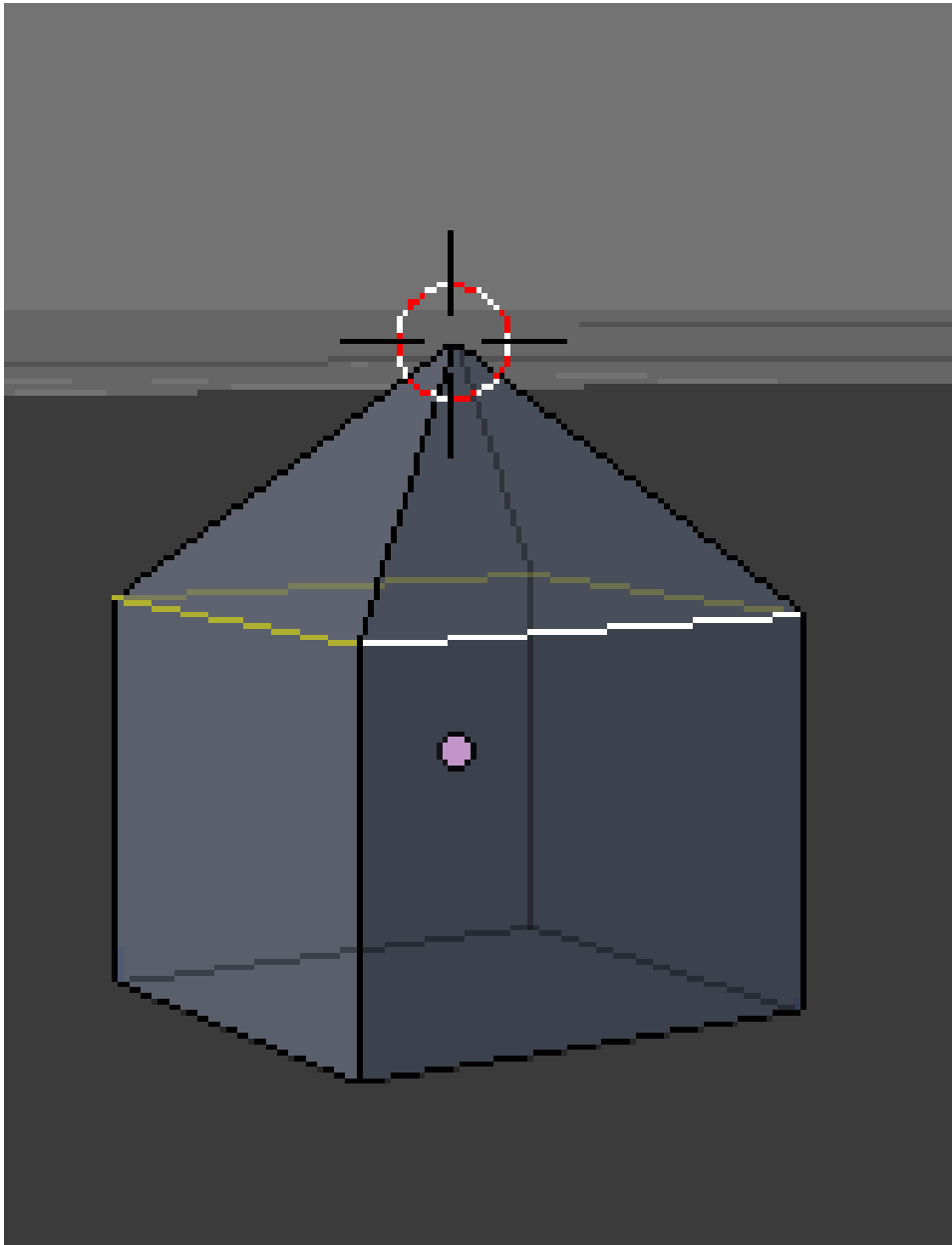


Figure 102 After step 2

Now extrude by scaling from that point:

1. Switch to Edge select mode.

2. Select just the four edges at the base of the roof.
3. Press **[E]** to activate the extrude tool.
4. Press **[S]** to extrude by scaling uniformly from the pivot point.
5. As you move the mouse pointer away from the pivot point, the roof of your house will expand.
6. When the roof is the size you want, confirm by **[LMB]** (or pressing **[ENTER]**).
7. Make the overhangs slanted by holding **[LMB]** on the blue arrow that appears in the center of the house, and dragging down.

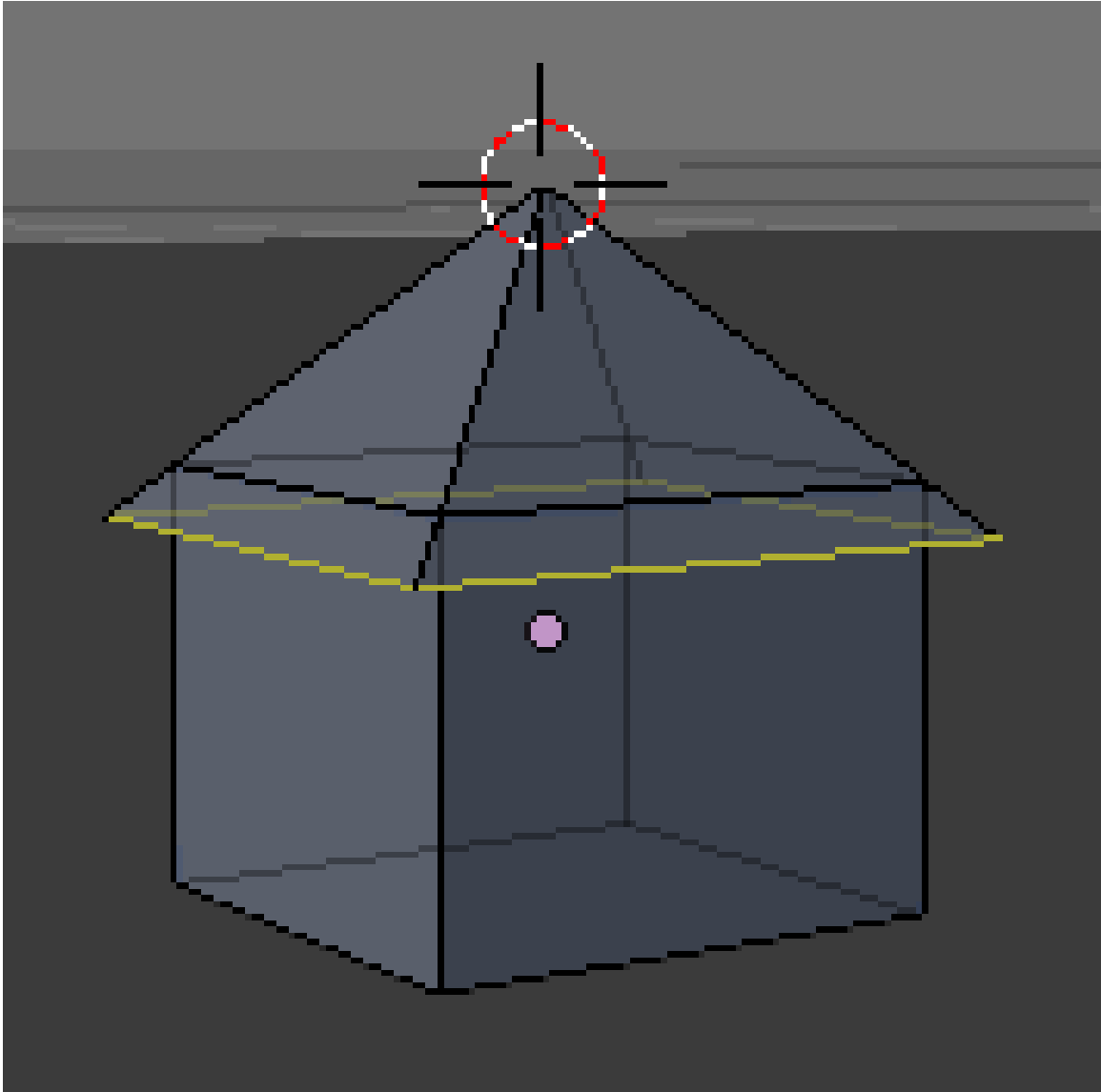


Figure 103 After step 7

29.5. Face Selection

It is often useful to select faces.

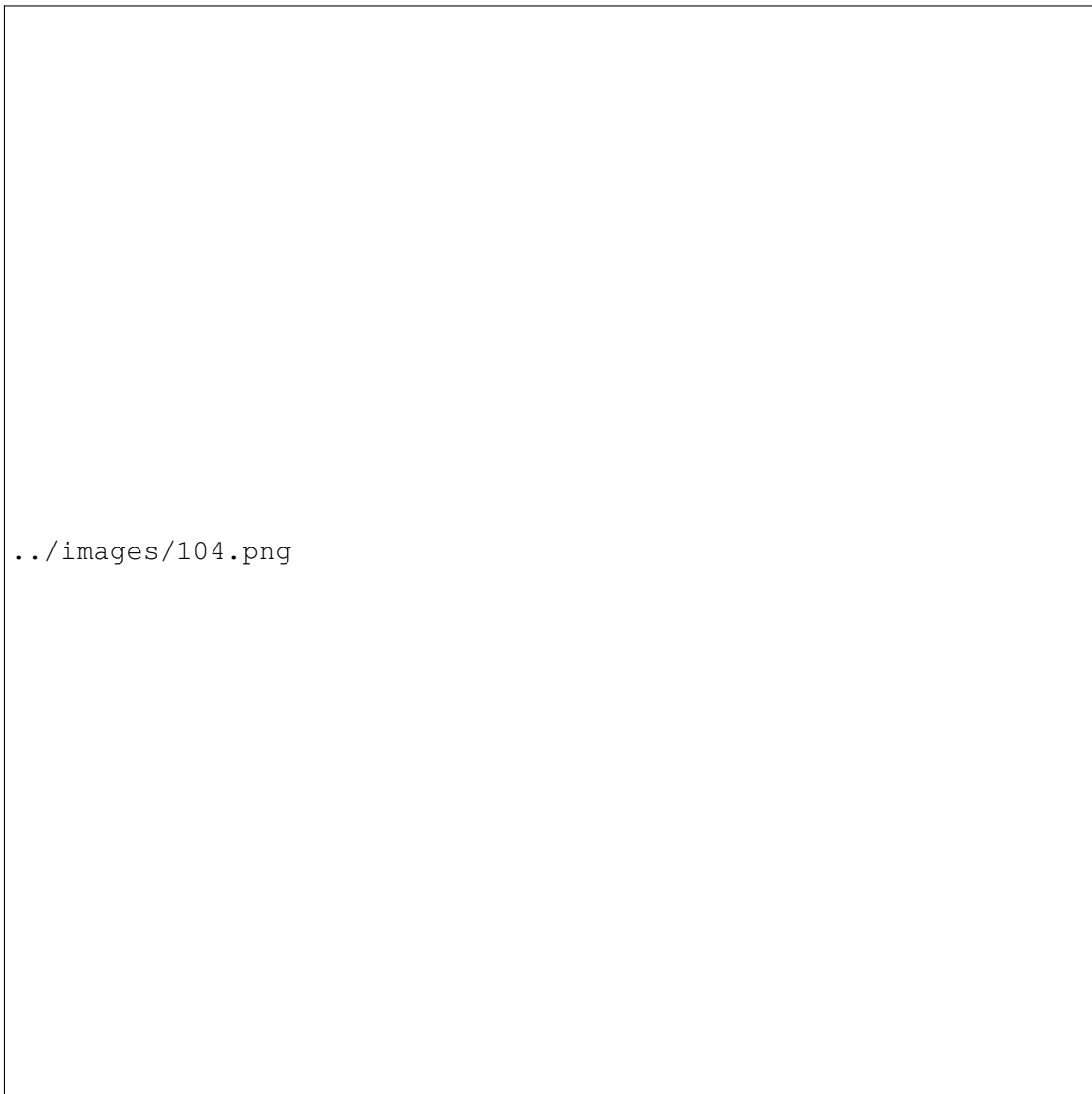








Figure 104 The select mode buttons after step 2

1. Make sure you're in Edit Mode on the house.
2. Click  on the Face select mode button in the 3D View header.

In Face select mode, the center of each face is marked with a small square. Faces appear as orange or stippled grey areas with orange edges when they're selected (depending on which face is active), and as grey areas when they're not.

Just as you selected edges in Edge select mode, you can now select (and deselect) faces:

- If any faces are selected, press  to deselect all faces.
- If no faces are selected, press  to select all faces.
- To select a single face (and deselect the rest), click  (or  + ) on the center of the face.

- To toggle the selection status of a face (without affecting the rest), click **SHIFT** + **RMB** on the center of the face.

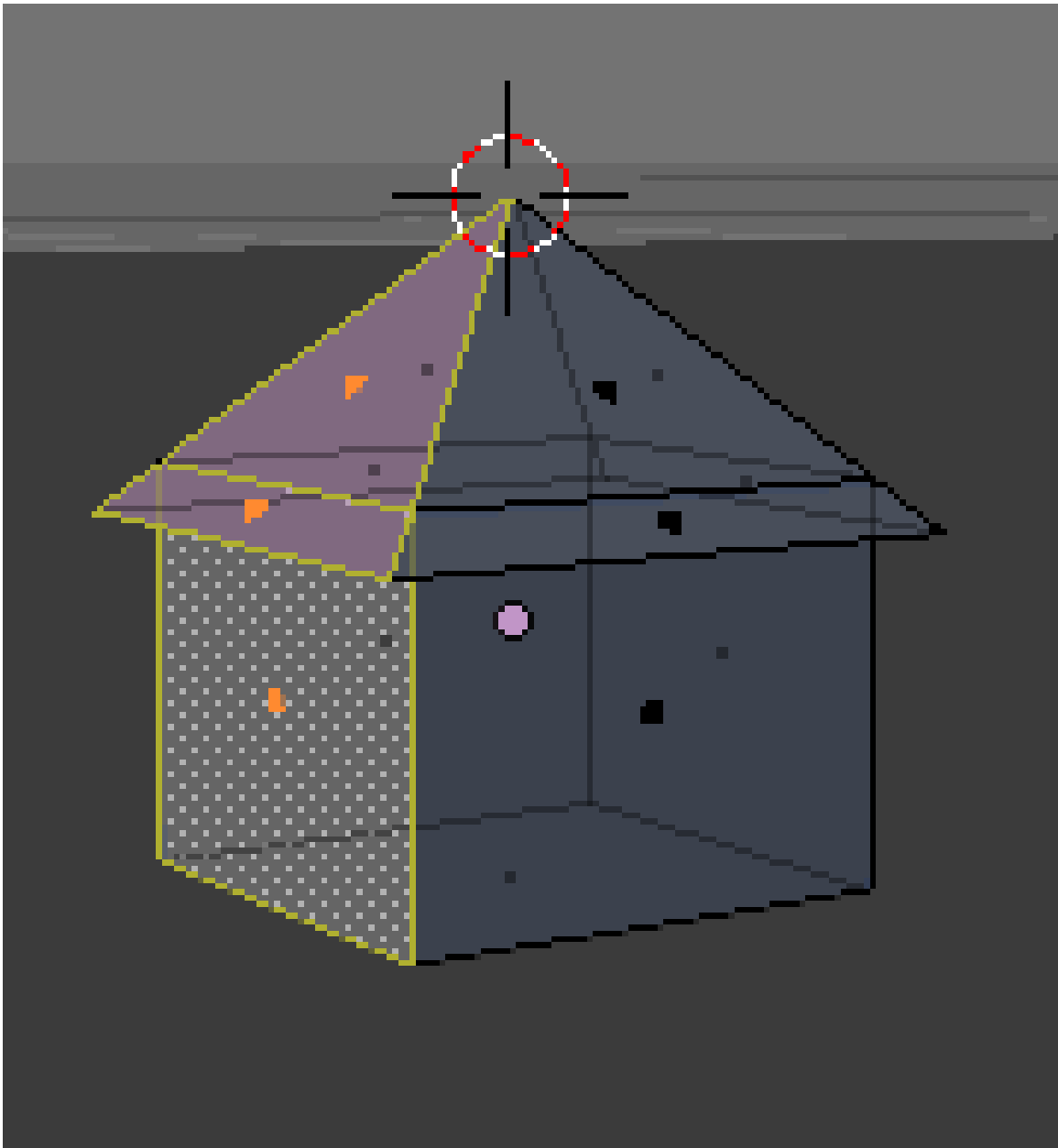


Figure 105 The three faces on the +X side, selected

Use these techniques to select all three faces (two roof and one wall) on the +X side of your house, as shown.

29.6. Extruding Faces

Just as you extruded edges to grow the roof, you can extrude faces to grow the entire house.

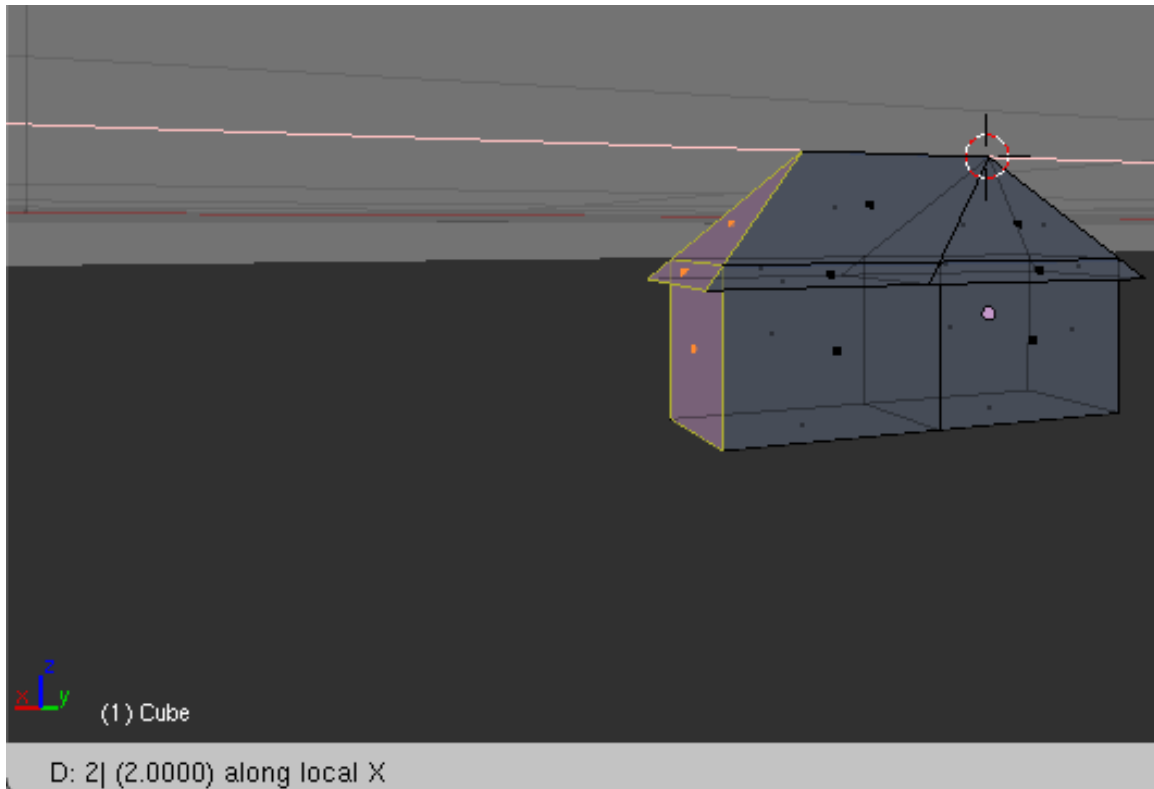


Figure 106 After step 6

To double the size of your house without changing the pitch of the roof:

1. With the three faces on the +X side selected, activate a 3D View window.
2. Press **E** to activate the extrude tool.
3. Press **X** to extrude along the X axis
4. As you move the mouse pointer in the +X direction, the +X half of your house will expand.
5. Press **2** to expand by exactly 2 Blender units.
6. Confirm and exit the extrude tool by clicking **LMB** (or pressing **ENTER**).

29.7. Deleting Edges

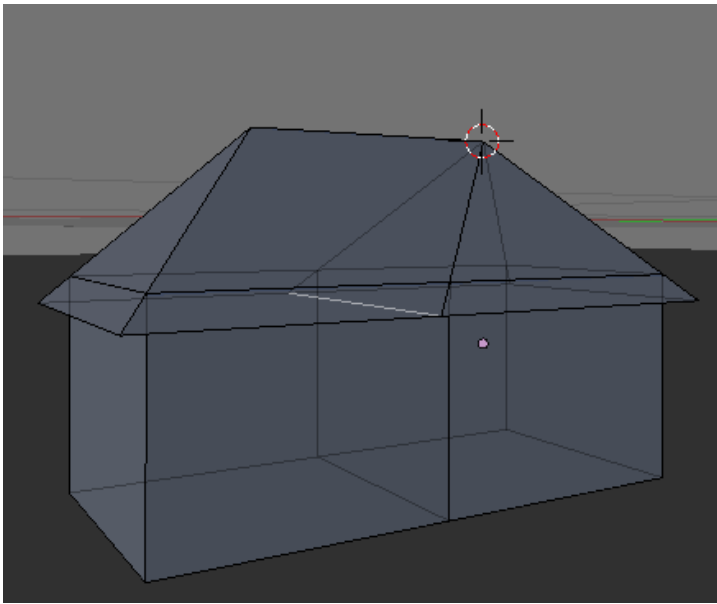


Figure 107 After step 2

If you look closely at the model, you'll notice an extra edge connecting the seams between the two halves of the roof. To delete this edge:

1. Edit the house object in Edge select mode.
2. Select just the edge you want to delete.
3. Press **X** or **DELETE**.
4. When the "Delete" menu comes up, choose *Edges*.

Note:

Deleting an edge automatically deletes any face(s) that include that edge.

29.8. Subdividing Faces

In order to add openings such as doors or windows to the walls of your house, you'll need to subdivide the wall (vertical) faces into smaller faces.

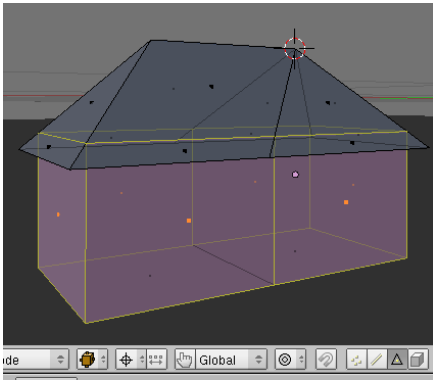


Figure 108 After step 2

To subdivide each wall face into a 10x20 grid:

1. Make sure you are not in wire-frame mode (otherwise the occlude hidden geometry button will not appear)
2. Edit the house object in Face select mode.
3. Select all six wall faces of your house.
4. Press **W** to bring up the *Specials* menu.
5. Choose *Subdivide*.
6. Set the number of cuts to 9 in the Operator panel (also accessible through F6).

You might be wondering why make 9 cuts instead of 10, the reason is that there is already (in a sense) one cut made with the face, so you only have to make 9 cuts. If you count the number of faces on the subdivided walls, you will find a 10x20 grid. The reason why there are 20 faces lengthwise is because you doubled the size of the house along the X axis (lengthwise).

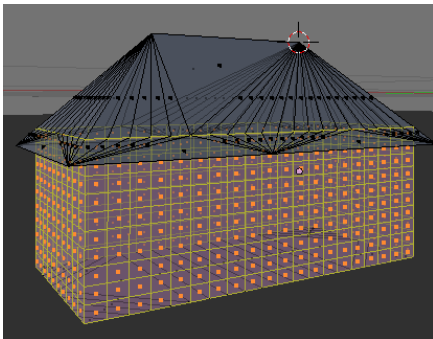


Figure 109 After step 6



Figure 110 Step 2

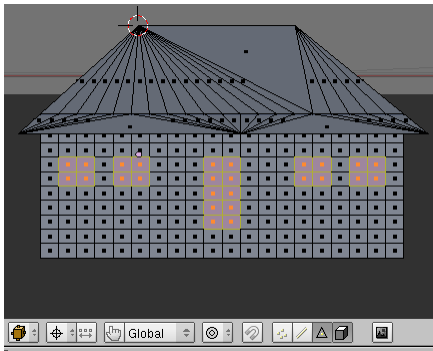


Figure 111 After step 3.2

Now you can extrude windows and doors:

1. Edit the house object in Face select mode.
2. Turn on the "Limit selection to visible (clipped with depth buffer)"(for old Blender versions "Occlude background geometry") option by clicking **LMB** on the toggle button in the 3D View header.
3. For each wall of the house:
 - a) Go to the perfect view for that wall:
 - **NUM1** for "front"
 - **CTRL** + **NUM1** for "back"
 - **NUM3** for "right"
 - **CTRL** + **NUM3** for "left"
 - b) Select faces where you want to create a window or door. An easy way to do this is by:
 - i. Deselecting all faces by pressing **A** once or twice.
 - ii. Pressing **B** to activate the Border Select tool.
 - iii. Clicking and dragging **LMB** to delimit a rectangular area.
 - iv. After you release **LMB**, all faces in the rectangular area will be selected.
 - c) Press **E** to activate the extrude tool.
 - d) Extrude inward 1/10th of a BU by typing **-1** and confirming it with **ENTER** or **LMB**.

29.9. Final Steps

1. Adjust the position of the lamp and aim the camera until you obtain a good render.
2. Save your work!

30. Modeling a Simple Person

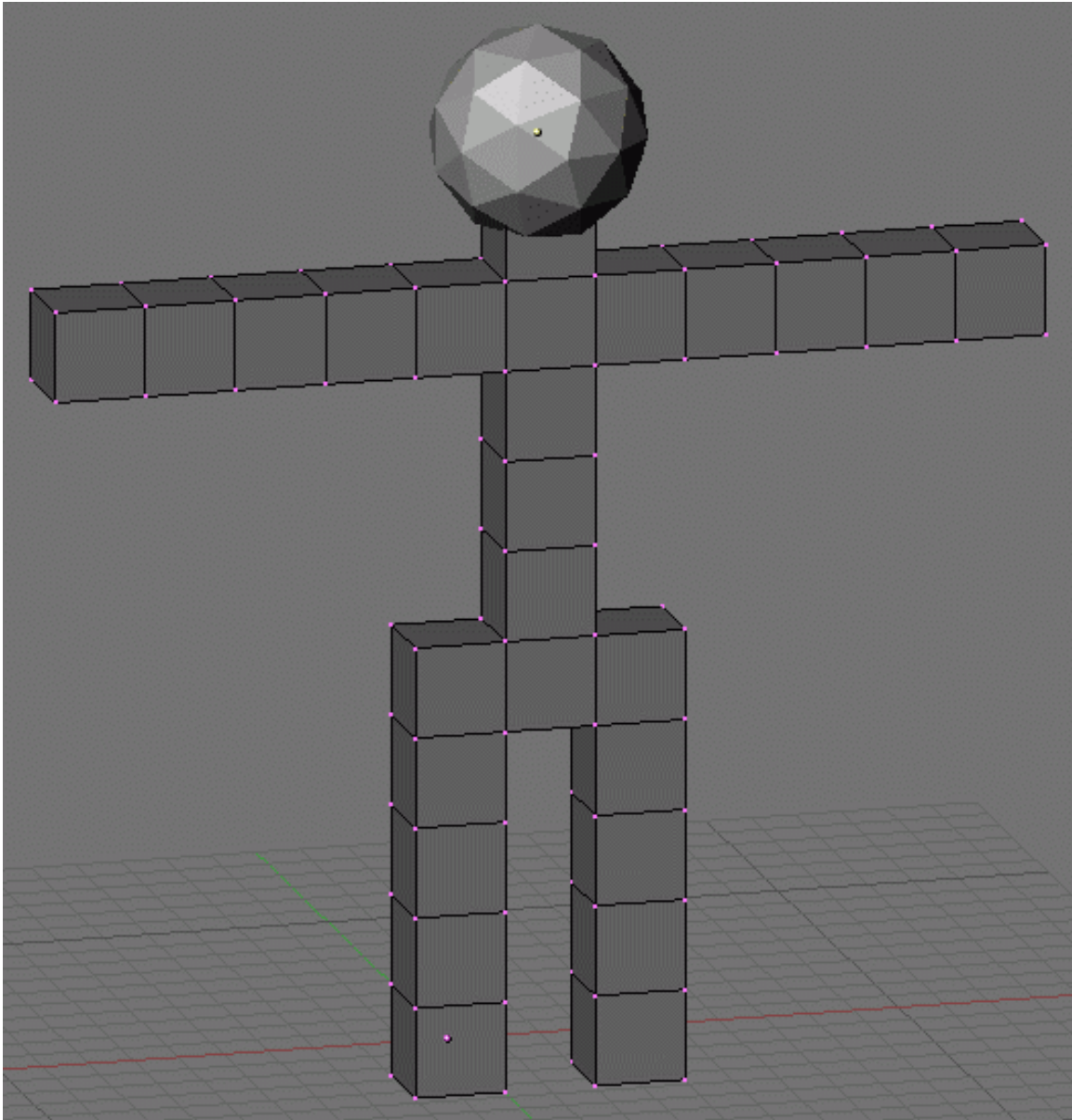


Figure 112 Your simple person will look like this.

In this module, you will model a simple human figure. Along the way, you will practice using extrusion and learn additional ways to select vertices, edges, and faces.

30.1. Start a New Scene

1. Start with the default cube (*File* → *Load Factory Settings*) and NumLock "on".
2. Press **TAB** to edit the cube.
3. Scale the cube down by pressing **S-5**.

30.2. Selection Methods

Just as you did for the house model, you will begin by selecting the top four vertices of the cube. This section presents six methods for doing so.

Ease of selection depends partly on the viewport settings and viewpoint. For greatest ease, you want a view in which the parts you are trying to select are both visible and close together.

For clarity, use a view of the cube in which all vertices are visible:

- Go to right side view with **NUM3**.
- Go to perspective view with **NUM5**.
- Disable the manipulator widget with **CTRL** + **SPACE**.
- Make sure the Occlude background geometry option is "off".

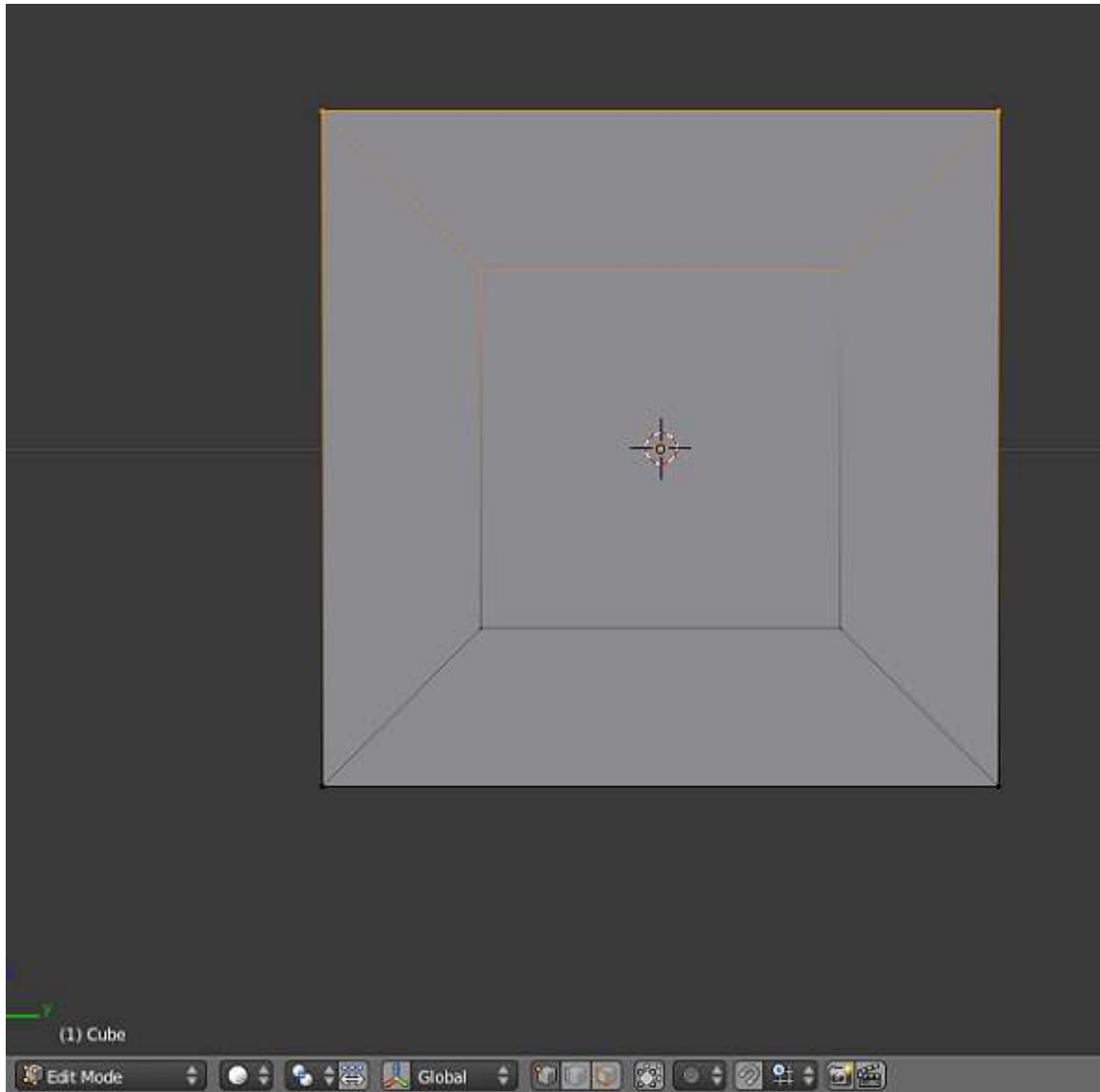


Figure 113

The picture on the right shows the cube with the correct vertices selected.

To begin, make sure you start in Vertex select mode.

30.2.1. Border Select Tool

The **border select tool** selects things that lie in a rectangular region of the viewport.

1. Activate (place the mouse pointer in) a 3D View window.
2. Deselect all vertices by pressing **[A]**.
3. Press **[B]** to activate the border select tool. Two dashed gray lines should appear, one vertical and one horizontal, forming a crosshair in the viewport.

4. Click and drag **LMB** diagonally across the area you want to select. The area will be outlined in dashed gray lines.
5. When you release the mouse button, the vertices inside the rectangle will be added to the selection.

Practice selecting the top four vertices this way. If you make a mistake, press **A** and try again.

30.2.2. Circle Select Tool

The **circle select tool** selects or deselects things that lie in a circular region of the viewport.

1. Activate a 3D View window.
2. Deselect all vertices by pressing **A**.
3. Press **C** to activate the circle select tool. A dashed gray circle should appear. note: Prior to Blender 2.5 **B** **B** twice.

When this tool is active, you can do various things:

- To move the select area, simply move the mouse pointer.
- To resize the select area, use **SCROLL** or **NUM+** / **NUM-** ..
- To select all vertices within the circle, click **LMB**.
- To deselect all vertices within the circle, click **MMB** or **ALT** + **LMB** or **CTRL** + **ALT** + **LMB**.
- To deactivate the tool, press **ESC** or **RMB**.

Practice selecting the top four vertices this way. If you make a mistake, press **A** and try again.

30.2.3. Lasso Select Tool

Like many graphics programs, Blender 3D has a **lasso select tool**.

1. Activate a 3D View window.
2. Deselect all vertices by pressing **A**.
3. Click and hold **CTRL** + **LMB**.
4. Drag the mouse pointer in a loop around the vertices you want to select. As you drag, a dashed gray line will appear.
5. You can deselect with lasso by pressing **CTRL** + **SHIFT** + **LMB**.
6. Release the **LMB** when you're done.






30.2.4. Vertex by Vertex Selection

You can select (or deselect) vertices one by one, as you did in the "Quickie Model" module .

1. Click **RMB** on a vertex to make it the only selected vertex.
2. Toggle the select state of additional vertices by clicking **SHIFT** + **RMB**.

30.2.5. Edge Select Mode




You can select (or deselect) edges one by one, as you did in the "Improving Your House" module .

1. Click  on the Edge select mode button in the 3D View header.
2. Select the top left edge of the cube by clicking on it with .
3. Toggle the select state of top right edge of the cube by clicking on it with  + .
4. Switch back to Vertex select mode by clicking  on the Vertex select mode button in the 3D View header.

After you switch back to Vertex select mode, all four vertices in the two selected edges are selected.

30.2.6. Face Select Mode

You can select (or deselect) faces one by one, as you did in the "Improving Your House" module .

1. Click  on the Face select mode button in the 3D View header.
2. Select the top face of the cube by clicking on its center dot with .
3. Switch back to Vertex select mode by clicking  on the Vertex select mode button in the 3D View header.

After you switch back to Vertex select mode, all four vertices in selected face are selected.

30.3. Extruding Limbs

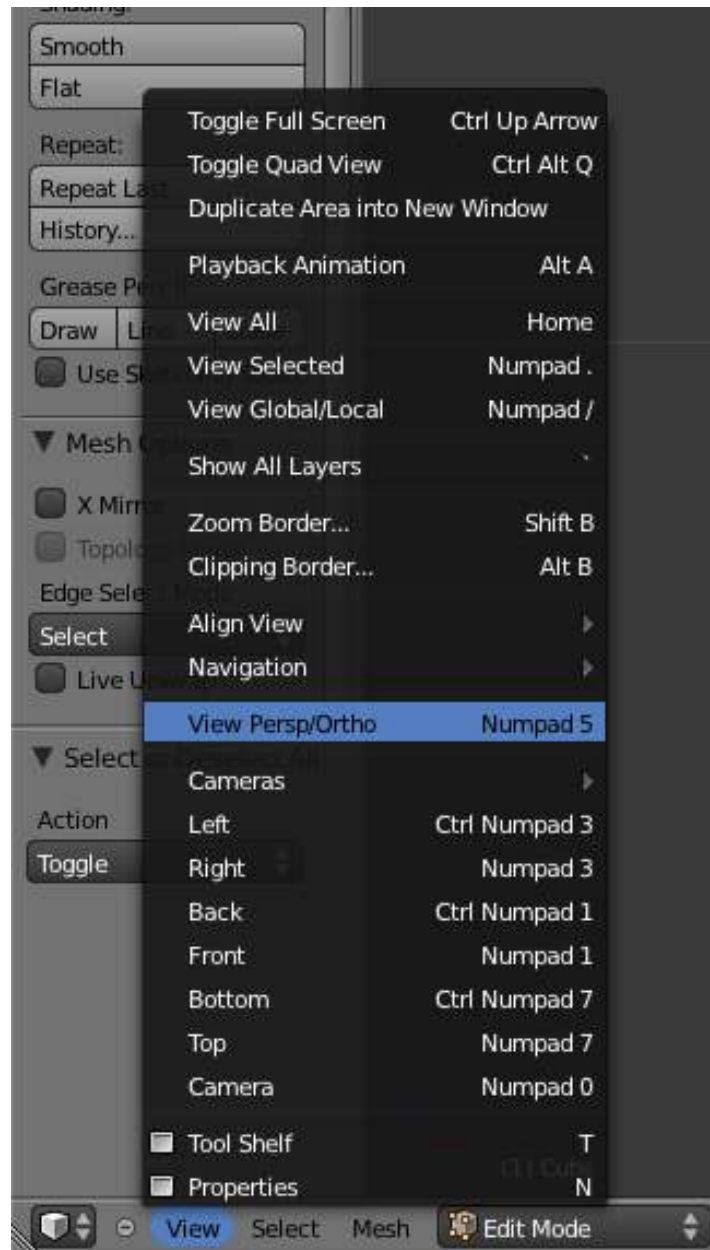


Figure 114 The view menu.

The illustrations in this section are in front orthographic view, so:

- Use **NUM5** (or *View* → *Orthographic*) to switch to orthographic view.
- Use **NUM1** (or *View* → *Front*) to switch to front view.

30.3.1. Region Extrusion

1. Make sure you're still in Edit Mode, with the top four vertices selected. (Only two will be visible in front ortho view.)
2. Activate the extrude tool by using **E** (or *Mesh* → *Extrude Region*).
3. Move the mouse pointer upwards. As you do, four *new* vertices will appear, each connected to one of the four that were previously selected.

The new vertices and their associated edges will move with the mouse pointer. You can lock them into place with **LMB** or **ENTER**).

30.3.2. Extruding a Leg

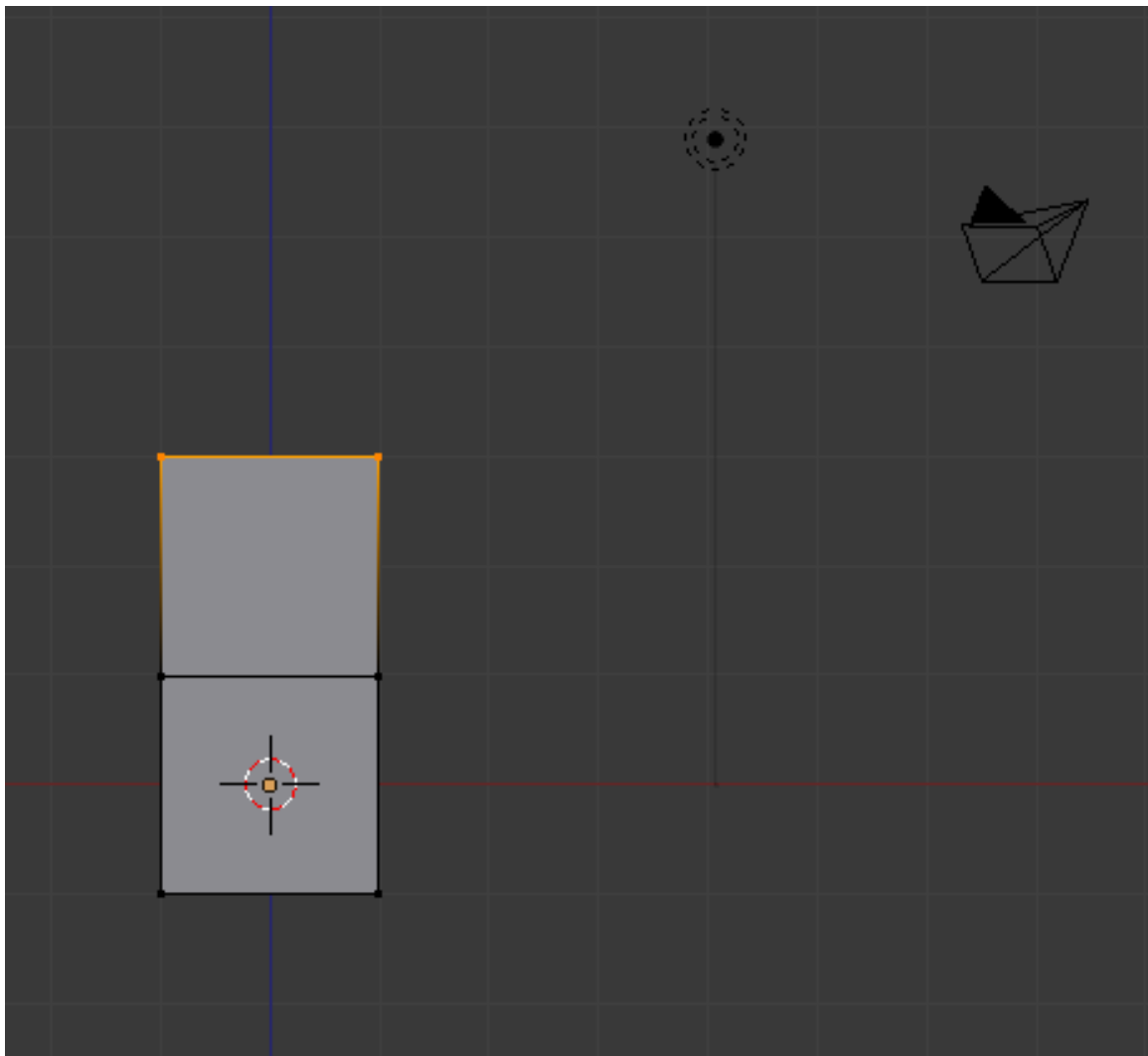


Figure 115 frame

Suppose you want to extrude a region the same size as the default cube -- in other words, one Blender units on a side.

1. Undo your previous extrude by pressing **CTRL** + **Z** .
2. Activate the extrude tool again by using **E** (or *Mesh* → *Extrude Region*).
3. This time, as you're moving the extruded vertices around, hold down the **CTRL** key. You'll see that the new vertices will only move in multiples of a Blender unit. This is called **snapping**, and it makes it easy to extrude by exactly one blender unit.

Continue extruding until you have five cubes of equal size stacked atop one another. This will be one leg of your figure.

Note:

Another way to extrude by exactly one Blender unit is to press **1KEY** while the tool is active. If you press **2KEY** when no tool is active, Blender will switch to the second layer, and your (first-layer) object will disappear. To make it visible again, press **1KEY** .

Note:

If you are not using Front Ortho view, the blender unit will be much larger than the cube. Switching to that view will allow for the proper size, although you can manually enter the extrusion as 0.4 units.

Note:

Don't extrude any cube more than a unit at a time. You'll want those extra verts, edges and faces later in this tutorial.

Note:

If the mesh gets too big for your view, you can zoom out using **SCROLL** or **NUM-**

30.3.3. Extruding the Pelvis

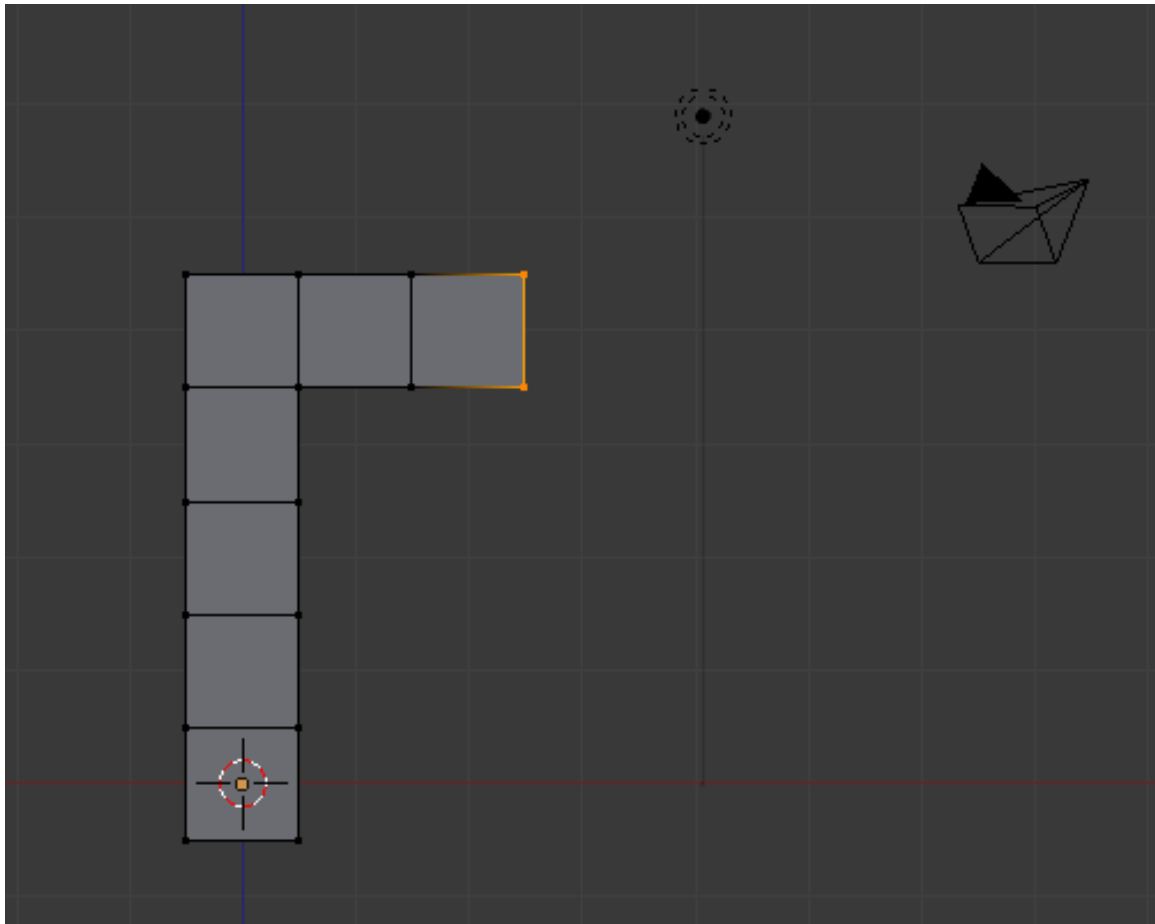


Figure 116 frame

1. Press **A** until all vertices are deselected.
2. Rotate the view (by dragging **MMB**) so you can see all four vertices on the right face of the top cube.
3. Select those four vertices.
4. Extrude twice to the right.

30.3.4. Extruding the Rest of the Body

The same trick is repeated over and over to build the rest of our simple body.

Note:

To speed things up, you may want to switch to Face select mode. In Face select mode, you can select a face with a single click.

1. Create a second leg by extruding down four times from the last cube of the pelvis.
2. Create the torso by extruding up five times from the middle cube of the pelvis.

3. Extrude to each side from the next-to-top cube of the torso to create arms.

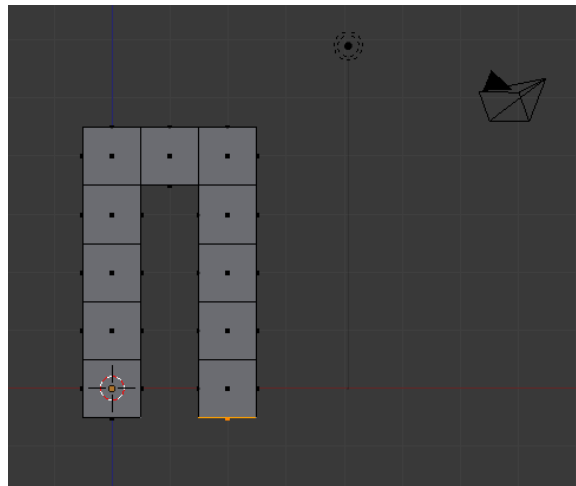


Figure 117 Legs and Pelvis

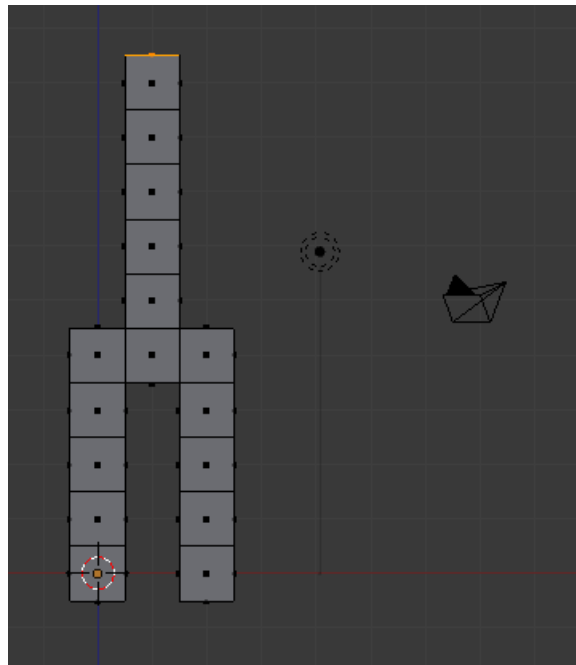


Figure 118 Torso

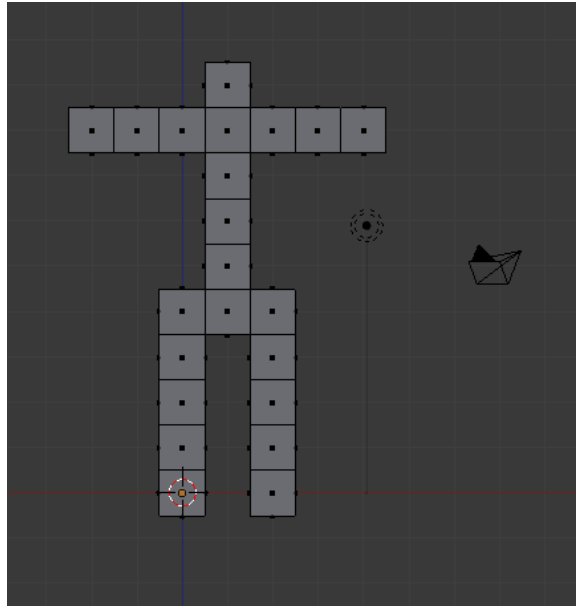


Figure 119 Arms

To be safe, remove any double vertices you may have inadvertently created:

1. In Vertex select mode, press **[A]** until all vertices are selected.
2. With a 3D View window active, press **[W]** to bring up the *Specials* menu.
3. Choose *Remove Doubles*.

Now check your work:

1. Return to Object Mode by pressing **[TAB]**.
2. Make sure the viewport draw type is Solid. (Press **[Z]** if it isn't.)
3. Rotate the viewpoint and examine the body from every side (it might be useful to return to perspective view for this).

This is easily fixed. To create a face:

1. Press **[TAB]** to go back into Edit Mode.
 2. Select four vertices.
 3. Press **[F]** (or choose *Mesh* → *Faces* → *Make Edge/Face* from the 3D View header).
- Note that you can also make edges with this tool if you select two vertices.

30.4. Adding the Head

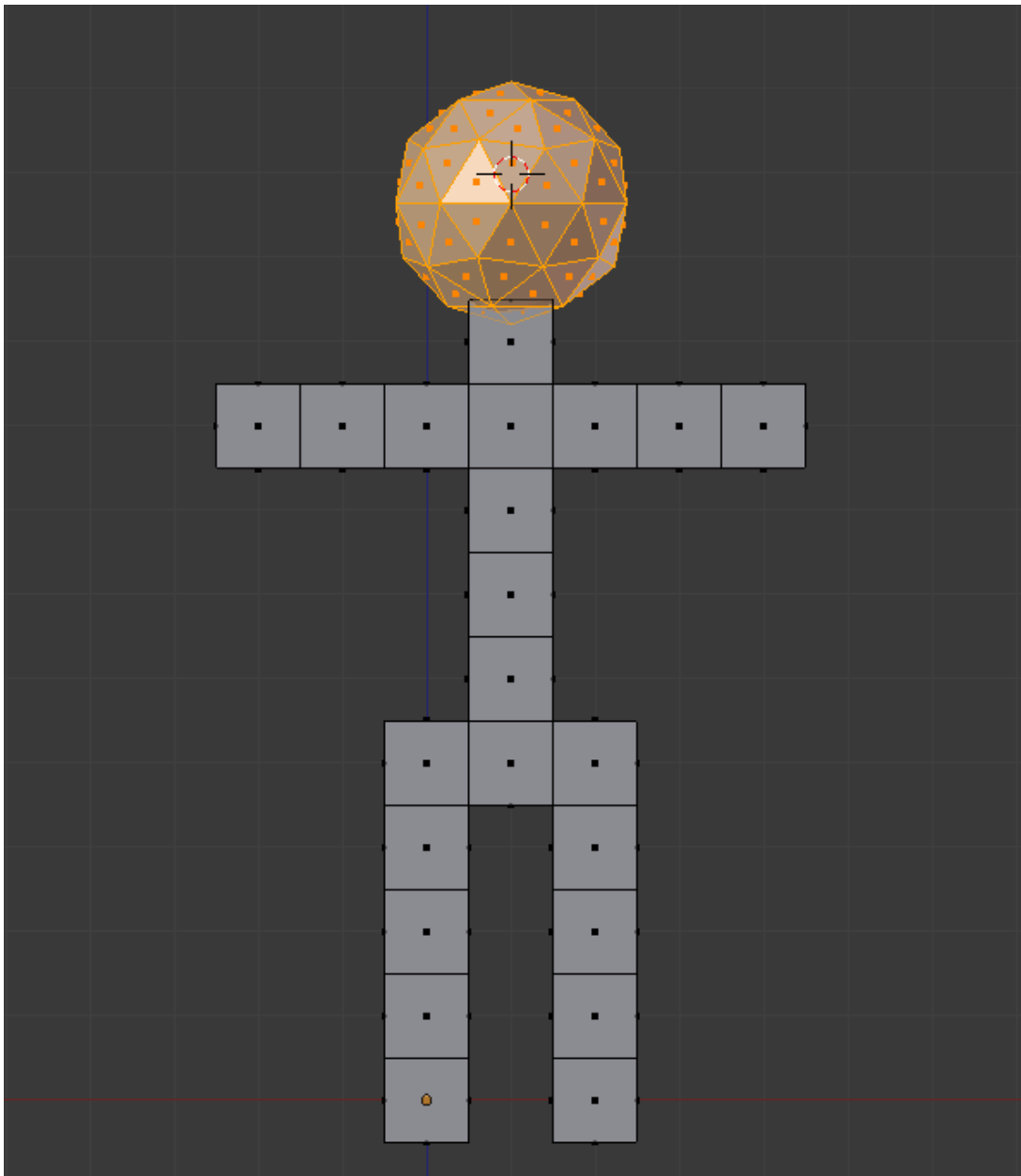


Figure 120 frame

1. Move the 3D cursor to a point above the neck by clicking with **LMB**.
2. Adjust the cursor position in orthographic top, front and side views (**NUM7**, **NUM1**, and **NUM3** respectively) until the 3D cursor is about where the center of the head should be. It may help to use **SHIFT** + **S** → *Snap* → *Cursor to Grid*.
3. Make sure you're in Edit Mode with a 3D View window active. (If you create the head in Object Mode, it will be a separate object from the body, and changes to the body later in this tutorial won't affect the head.)

4. Create a sphere using **SHIFT** + **A** → *Mesh* → *Icosphere*.
5. Leave the default settings for subdivisions and size in the bottom left of the screen.

You should now have a small sphere at the top of the body. To make it more proportional to the body, resize it using the scale tool:

1. Make sure you're still in Edit Mode, with a 3D View window active and the head selected.
2. If necessary change the pivot point to *Median Point*.
3. Activate the scale tool by pressing **S** (or *Mesh* → *Transform* → *Scale*).
4. Move the mouse pointer until the head is the size you want.

You may also adjust its position using the grab tool:

1. Make sure you're still in Edit Mode, with a 3D View window active and the head selected.
2. Activate the grab tool by pressing **G** (or *Mesh* → *Transform* → *Grab/Move*).
3. Move the mouse pointer until the center of the head is where you want it.

Note:

If you deselect the head and then decide that you want to select it again:

1. Hover the mouse pointer over any vertex/edge/face.
2. Press **L**. In Edit Mode this will select all vertices that are linked to the vertex nearest the mouse pointer.

Now check your work:

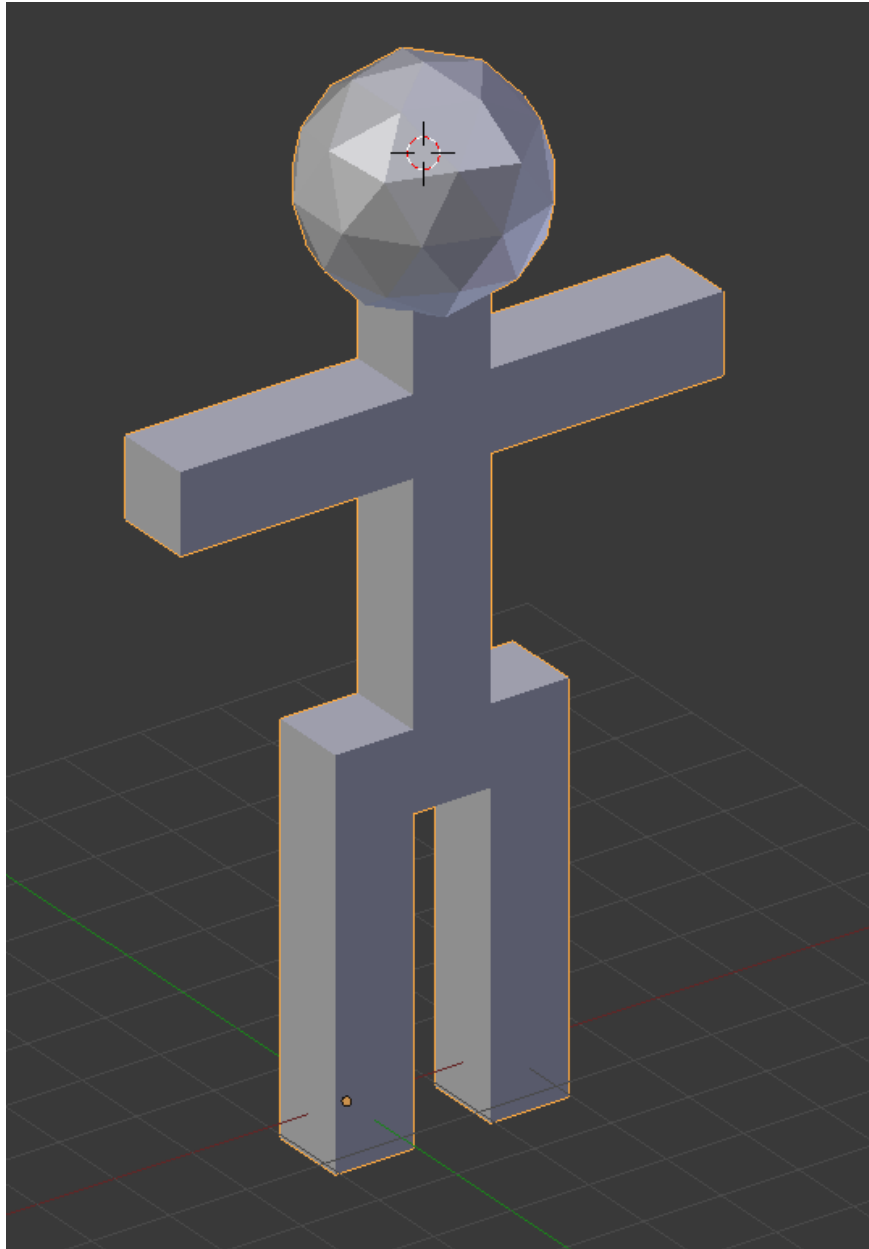





Figure 121 frame

1. Return to Object Mode by pressing **[TAB]**.
2. Make sure the viewport draw type is Solid. (Press **[Z]** if it isn't.)
3. Rotate the viewpoint and examine the body from every side. Make sure that the head connects properly to the neck.

30.5. Save Your Work

You will continue working on your simple person model in the next module.

To save the scene in a `.blend` file:

1. Press  +  (or select *File* → *Save*).
2. Navigate to the directory (folder) where you want to write the file.
3. Type a filename in the text box to the left of the "Cancel" button.
4. Click  on the "Save Blender File" button.

31. Detailing Your Simple Person 1

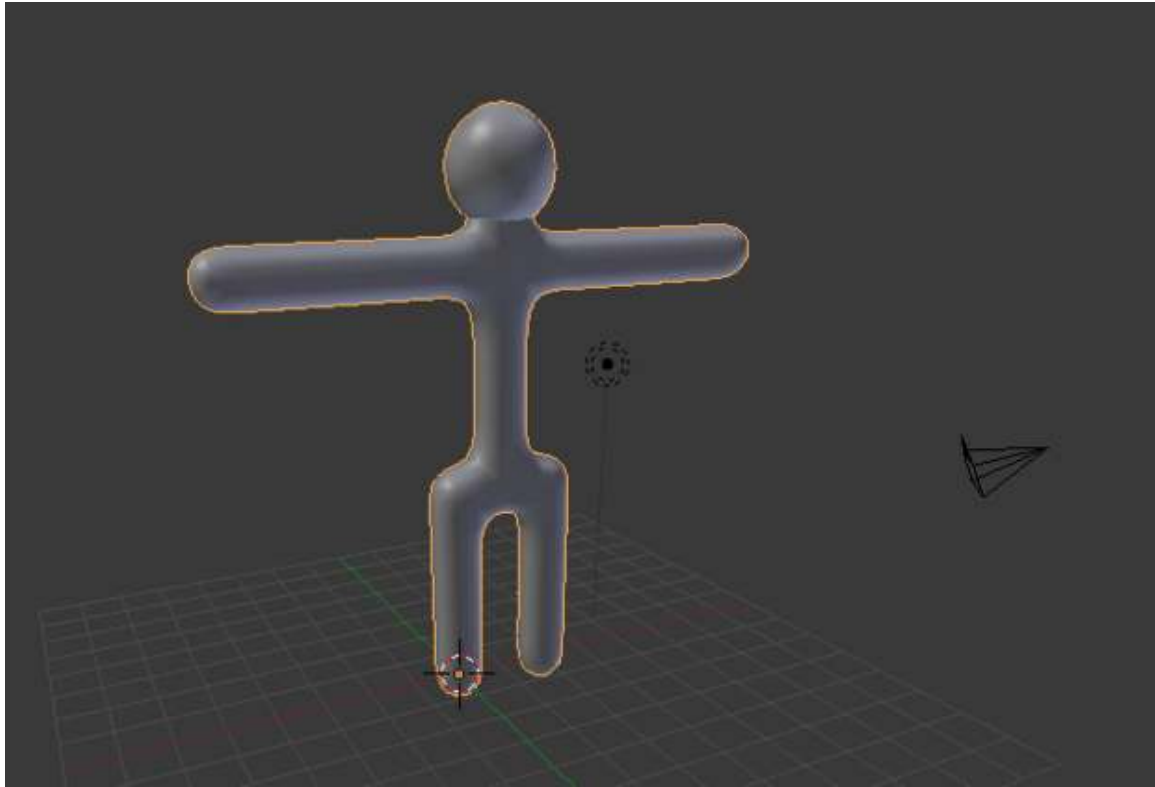



Figure 122 Your goal.

Few real-life objects have perfectly sharp edges. People, in particular, consist of mainly smooth surfaces. How does one model a smooth object using flat faces and sharp edges?

In this module, you'll learn how to smooth a mesh by using subsurfaces and smooth shading.

You'll need the simple person model from the previous module. If you haven't done it, either go back and do it now or download the pre-made model from Yosun Chang's website at [.](#)

If the model doesn't look solid, your Viewport Shading setting may be set to Wireframe. To switch to Solid shading:

1. Activate the 3D View window.
2. Press .

31.1. Subsurfaces

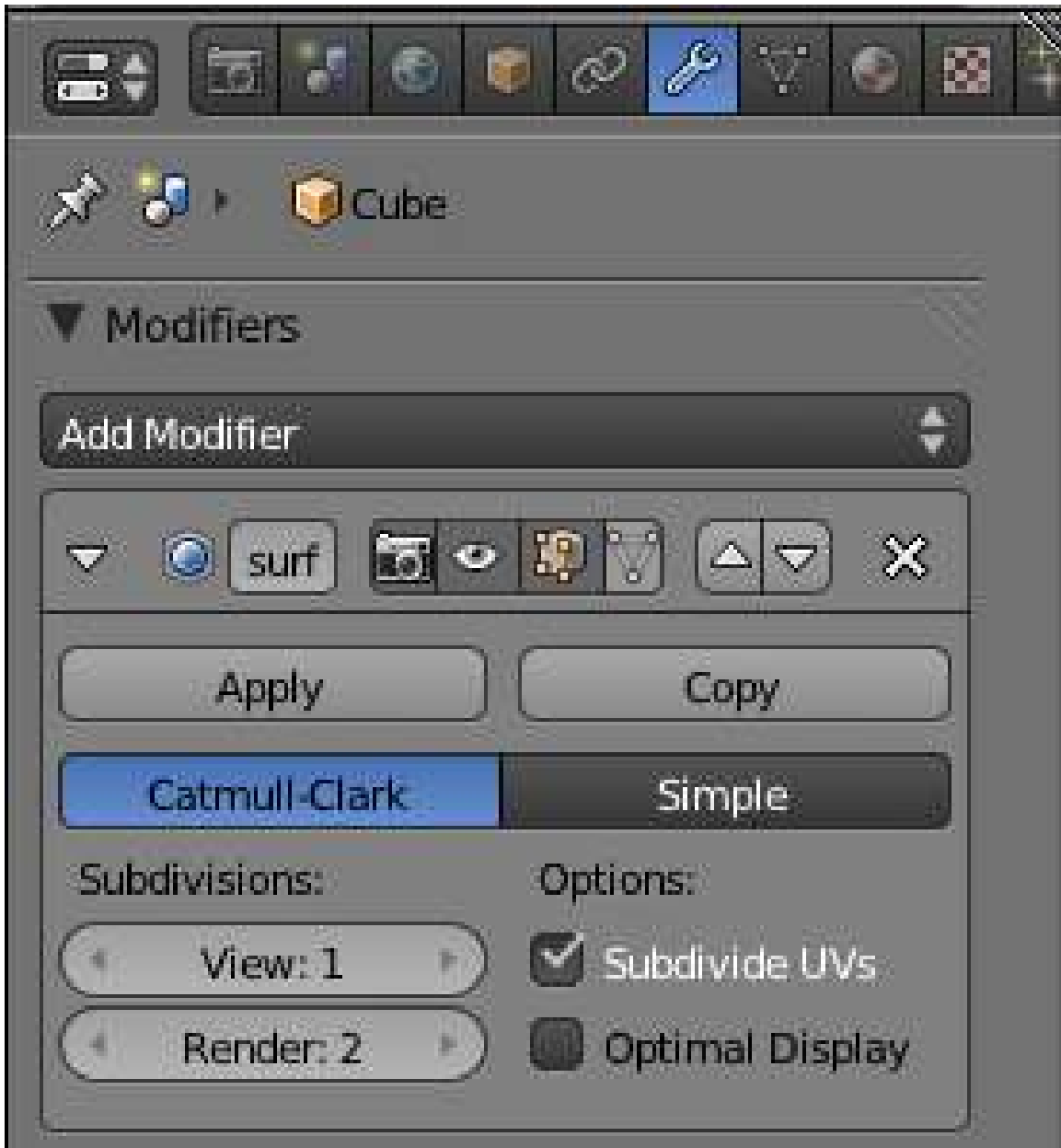


Figure 123 The sub surface modifier.

So far, all the meshes you've created have had sharp edges, giving them a faceted appearance like that of a cut diamond. To model a smooth object (like a human body) you might think you'd need a huge number of vertices and faces. **Subsurfaces** partly solve this problem by automatically subdividing a mesh into a finer mesh suitable for smooth rendering.

You subsurface in Blender by adding a **subsurf modifier** to a pre-existing mesh object. A **modifier** is simply an algorithm (automatic process) which can be added to an object.

To get started, make sure Blender is in Object Mode, with only the simple person object selected:

1. If Blender is in Edit Mode, press .
2. To select the simple person,  on it.

To add a subsurf modifier to the selected object:

1. Click on the modifiers tab (wrench icon) in a Properties window.
2. *Add Modifier* → *Subdivision Surface*.

The object's appearance should immediately become more faceted and more rounded. In addition, several subsurface controls will appear in the Modifiers tab.

Note:

The modifier has been added, but it hasn't actually been "applied" yet.

The model may include some double vertices. To get rid of these:

1. Edit the model in Vertex select mode.
2. Select all vertices.
3. *Mesh* → *Vertices* → *Remove Double*
4. Try again.

You need to look in to upgrading (or possibly even downgrading) your graphics drivers. Having the right graphics driver can avert a lot of problems.



What just happened? The default subsurf modifier (one level of Catmull-Clark) subdivided each face of the object into four smaller faces that are progressively angled. This softened the sharp edges of the original model where faces met at 90-degree angles.

31.1.1. Controls

For this model, one level of subsurf isn't quite enough. To increase the number of levels to two, just increase the number in the text box directly underneath *Subdivisions*. The *View* setting controls the number of subdivision levels visible in the viewport. This can be very useful when you have a high-poly scene, just decrease the number of visible subdivisions to speed up viewport action.

You can specify additional levels of subsurfing to be used during renders. For extra smooth renders, you might want three levels of subsurfing. Set this with the *Render* control immediately below the *View* control.

The *Apply* button applies the modifier to the mesh. Do not click it yet; we'll be playing with the model a bit longer before we want to apply the changes. While useful with some modifiers, applying a subsurf modifier produces a very complex mesh, and there's no need to do so here.

Remember that you can undo any accidental modifications with  + .

Blender can combine a series of modifiers by **stacking** them. For this reason, the Modifier tab includes buttons for arranging and removing modifiers.

You can hide edges created by the modifier by activating the *Optimal Display* toggle button. The effect of Optimal Draw is especially clear with the Wireframe draw type.

You can edit the mesh in modified form (without actually applying the modifier) by activating the *Apply modifier to editing cage during Editmode* toggle button, a small button to the left of the

Up/Down arrows (the arrows are for changing the position of the modifier in the stack) in the Modifier panel.



Figure 124

Try this out:

1. Press **[TAB]** to enter Edit mode.
2. Make sure Blender is in Vertex select mode.
Note that the vertices no longer lie on the surface of the object.
3. Activate the *Apply modifier to editing cage during Editmode* button.
Now all vertices lie on the surface of the object, and you can adjust the (modified) vertices directly. However, any additional vertices created by the modifier cannot be directly edited without applying the modifier.

You will be editing the boxy version of the simple person awhile longer, so before continuing, deactivate the *Apply modifier to editing cage during Editmode* button.

31.2. Smooth Shading

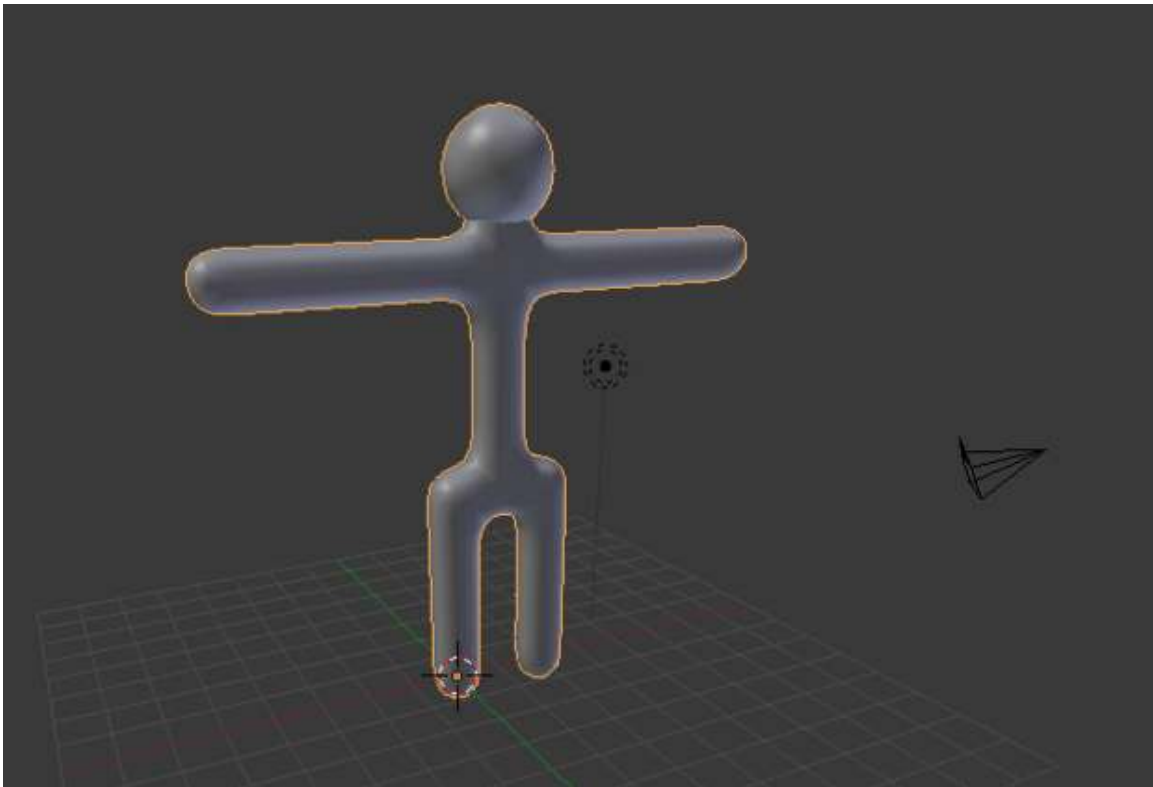


Figure 125 Your simple person after setting smooth.

Subsurfaces do a good job of smoothing out corners in meshes. Even with two levels of subsurfaces, however, the simple person does not look completely smooth; when viewed close up, it has a scaly appearance. This is because each face is **flat shaded**—shaded to resemble a flat surface—resulting in sudden changes in brightness at most edges. For a smooth object, you want **smooth shading**, which smooths out the changes in brightness.

1. Go to Object Mode.
2. Set the draw type of a 3D View window to "Solid".
3. Select your subsurfed object.
4. In the Toolshelf on the left, look for a caption called *Shading*. Under it should be a button called *Smooth*.
All the mesh edges will be smoothed out, leaving no sudden changes in brightness. The faces blend smoothly into one another, making the edges nearly invisible. If the icosphere has not smoothed properly and is dimpled, enter Edit Mode by pressing **[TAB]**, select all vertices (A) and recalculate the normal direction (CTRL+N). This is also available in the Toolshelf under *Normals*.
5. Click the other button under *Shading* in the Toolshelf, named 'Flat'.
The edges will reappear. Now you know the difference between Flat and Smooth.
6. Since the model looks better with smooth shading, click **[LMB]** on the "Smooth" button again.

Note that if you didn't have subsurf enabled, then the mesh wouldn't look much different. This is because smooth shading doesn't affect the mesh shape, it just changes how the computer draws the triangles.

Smooth shading also removes a lot of definition, a good way to get rid of this is simply to add a subsurf modifier like you just did. The modifier will not only require less vertices, but add definition.



Figure 126 Flat Shading



Figure 127 Smooth Shading

Save your work. You will continue refining this model in the next module.

31.2.1. Additional Resources

w:Flat shading¹ w:Gouraud shading² w:Phong shading³ w:Subdivision surface⁴

- For more about modifiers, see [The Blender Manual page on The Stack](http://wiki.blender.org/index.php/Doc:Manual/TheStack) at <http://wiki.blender.org/index.php/Doc:Manual/TheStack>
- For more about Blender subsurfaces, see [The Blender Manual page on Subdivision Surfaces](http://wiki.blender.org/index.php/Doc:Manual/SubdivisionSurfaces) at <http://wiki.blender.org/index.php/Doc:Manual/SubdivisionSurfaces>.

1 <http://en.wikipedia.org/wiki/Flat%20shading>

2 <http://en.wikipedia.org/wiki/Gouraud%20shading>

3 <http://en.wikipedia.org/wiki/Phong%20shading>

4 <http://en.wikipedia.org/wiki/Subdivision%20surface>

32. Detailing Your Simple Person 2

In this module, you'll edit a subsurfed mesh using the scale and grab tools, all the while improving your character.

You'll need the simple person model from the previous module. If you haven't done it, either go back and do it now or else download the pre-made model from Yosun Chang's website at [.](#)

32.1. Widening the Torso

To be realistic, the simple person's torso needs to be three times wider. In order to keep the torso symmetrical, you'll expand it by scaling both sides from a central point.

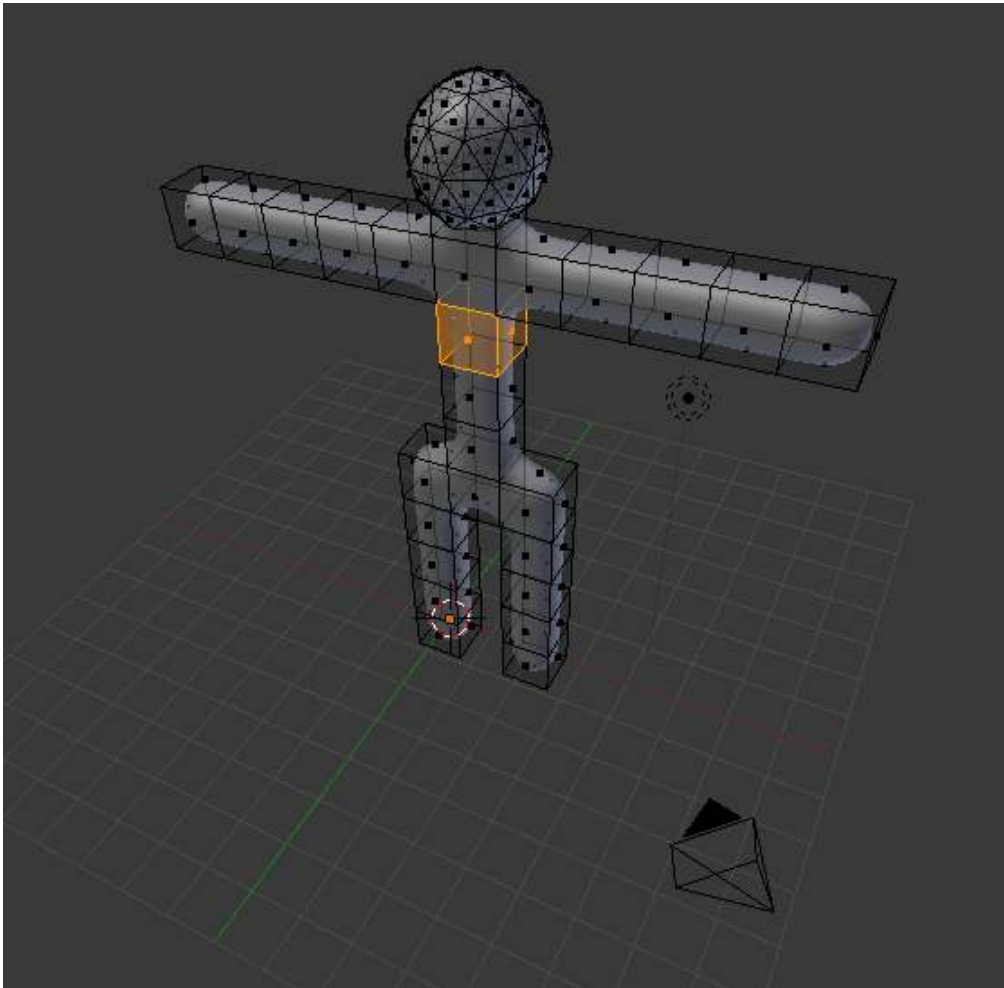


Figure 128

Select the sides of the torso:

1. Enter edit mode on the simple person.
2. In the 3D View header, set Face select mode.
3. From the 3D View header, choose *Pivot* → *Median Point*.
4. In the 3D View header, make sure Proportional Edit button is off.
5. Select the three faces on both the left and right sides of the torso, between the armpits and the waist.

Note:

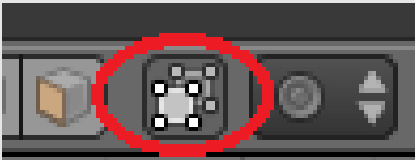


Figure 129 The cube icon toggles the visibility of certain components.

When editing in solid mode, the vertices, edges and faces on the back side of the model are, by default, invisible. This feature can be toggled by clicking **[LMB]** on the *Occlude Background Geometry* button in the 3D View header. Toggle it on and off a few times and observe how the back faces appear and disappear.

We will now scale the torso with the scaling tool:

1. Activate the 3D View window and press **[S]**, **[X]**.
2. Adjust the amount of scaling. Either:
 - Move the mouse pointer until the torso is the width you want.
 - or
 - Press **[3]**
3. Confirm and exit by pressing **[ENTER]** or clicking **[LMB]**.

Note:

Scaling faces causes adjacent edges and faces to move, due to their shared vertices. You cannot separate a face or edge from its vertices.

Continue selecting different parts of the torso and scaling them to get more practice using the above scaling methods.

32.2. Bending the Arms

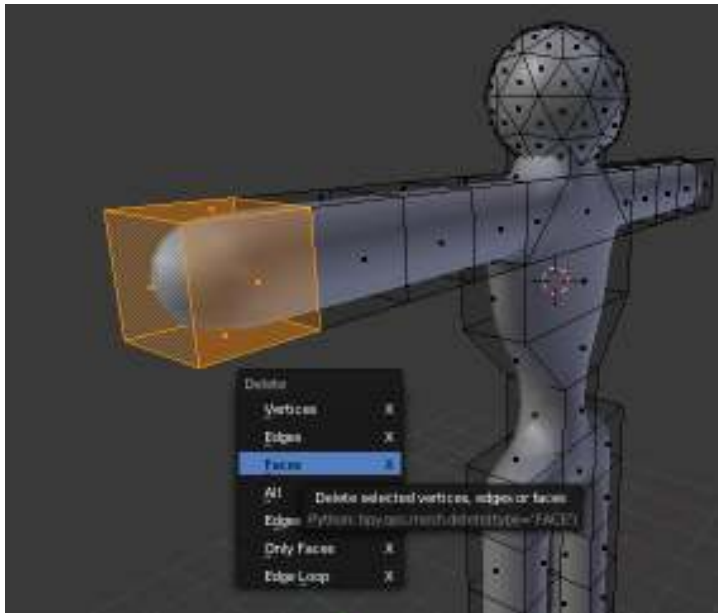


Figure 130 Removing the forearm

When you've got the basic shape of the torso, make the person hold up his hands. You'll do this by deleting the forearms and then extruding upward from the elbows.

Select both forearms:

1. Enter edit mode on the simple person.
2. In the 3D View header, set Face select mode.
3. With the 3D View window active, press **A** until all vertices are deselected.
4. Select the five faces at the end of the forearm.

Now erase them:

1. Press **X** to open the Delete menu.
2. Choose *Faces*.

The forearm will disappear, leaving a hole. Don't panic; we'll fix it later. Now to make the arm point upwards:

1. Select the top face of the last remaining "arm cube".
2. Extrude the region upward by two Blender units **E**, **Z**, **2** and confirm with **LMB** or **ENTER**.

The hole in the elbow is caused by a missing face. To fill in the missing face:

1. Deselect all vertices.
2. Select the four vertices surrounding the missing face.
3. With the 3D View window active, create the face using either
 - *Mesh* → *Faces* → *Make Edge/Face*
 or

- 

Note:

If a *Make Faces* menu appears when you try to fill the hole, it may be that you have some doubled vertices. Do "Remove Doubles" on the entire model and try again.

The new face should be smooth. If it isn't, make it so, using *Mesh* → *Faces* → *Shade Smooth*.

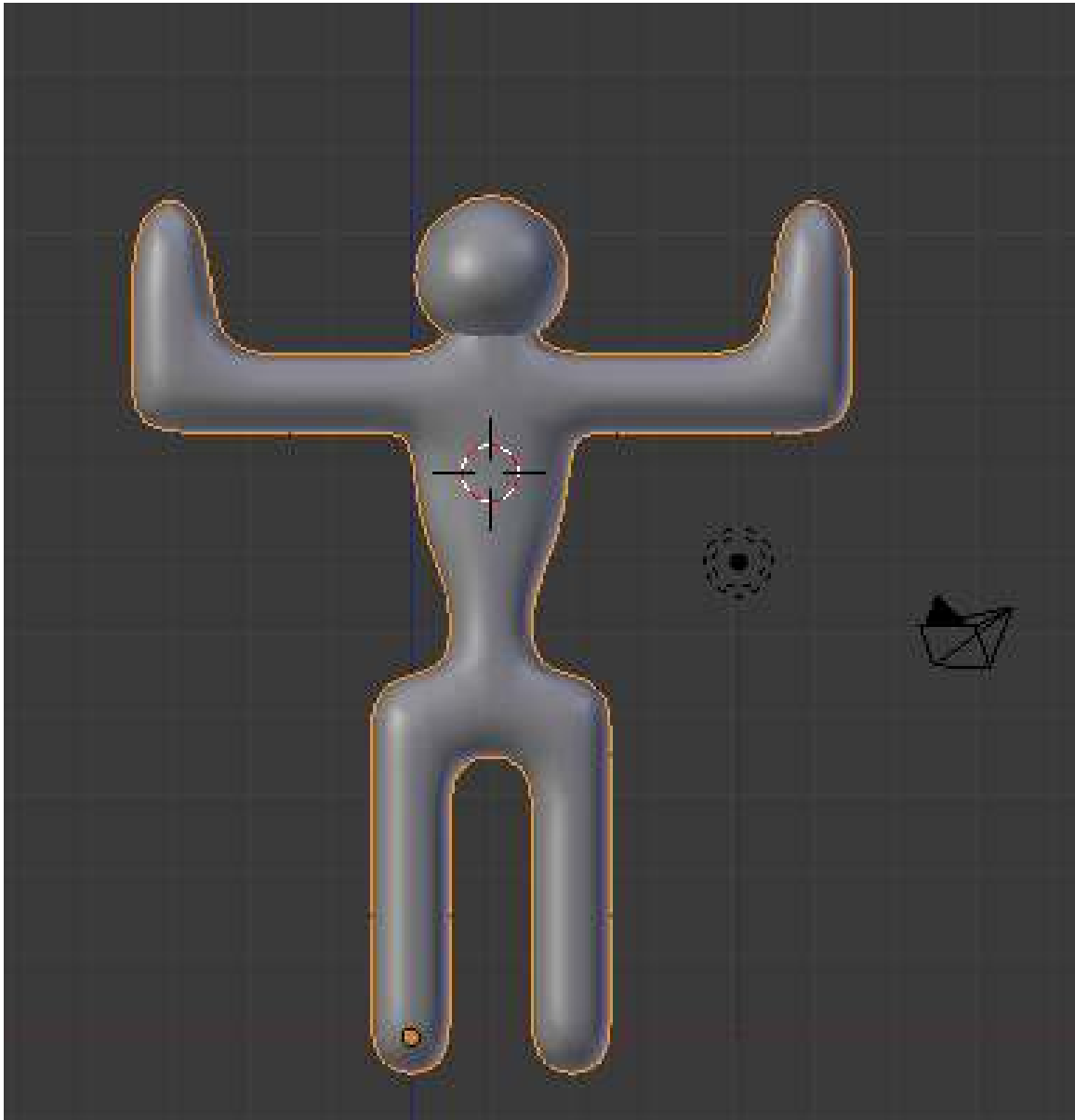


Figure 131 Repeat on the other side

Go through the same steps (erase, extrude, and fill) on the other arm. Be sure to deselect all vertices in the first arm before selecting any in the other arm. If you have difficulty making the arms symmetrical, undo your work and go through the steps simultaneously on both arms.

32.3. Making Feet

To make feet for your simple person, you subdivide the ends of the legs and pull the front edges forward.

1. Edit the simple person in Face select mode.
2. Select two bottom faces of the legs (front of the feet) by clicking **[RMB]** on the first and then **[SHIFT] + [RMB]** on the other.
3. Subdivide both faces, either with:
 - **[W]** *Subdivide*
 - or
 - *Mesh* → *Edges* → *Subdivide*

Each face gets subdivided into four smaller faces.

Now select the front edges and pull them forward:

1. Switch to Edge select mode.
2. Press **[A]** until no edges are selected.
3. Select the four bottom front edges of the soles (two for each foot) (where the toes should be).
4. Press **[G]** and limit movement to the Y axis.
5. Move the mouse pointer until the feet are the length you want.
6. Confirm and exit by pressing **[ENTER]** or **[SPACE]** or clicking **[LMB]**.

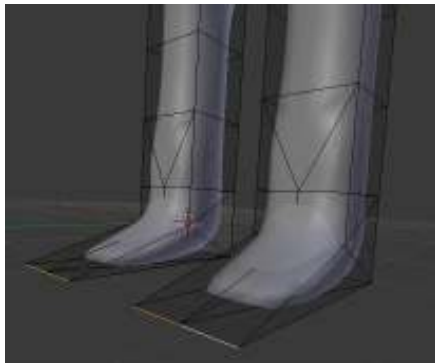


Figure 132 Congratulations! You now have feet.




32.4. Reshaping the Head

When you're satisfied with the torso and limbs, you should do something about that head. A bit too spherical, isn't it? You can elongate it by scaling along the Z axis.


When scaling the head, you want to make sure that it stays connected to the neck.

First, place the 3D Cursor at the base of the head, where it meets the neck. An easy way to do this is as follows:

1. Go into Vertex select mode.

2. Make sure the *Occlude background geometry* option is "off".
3. Select the vertex at the base of the head using .
4. Snap the cursor to this vertex using  +  *Cursor to Selected*


Now select the entire head:

1. Hover the mouse over a vertex/edge/face of the head
2. Press  to select all parts linked to that part.

Note:

This works even when the head and body meshes overlap, so long as they aren't linked together anywhere.

Tell Blender that you want to pivot around the 3D Cursor by changing the pivot point to *3D Cursor* on the Pivot menu.

Now scale the head along the Z-axis, using either the scale tool () , scaling by 1.5 should be about right.

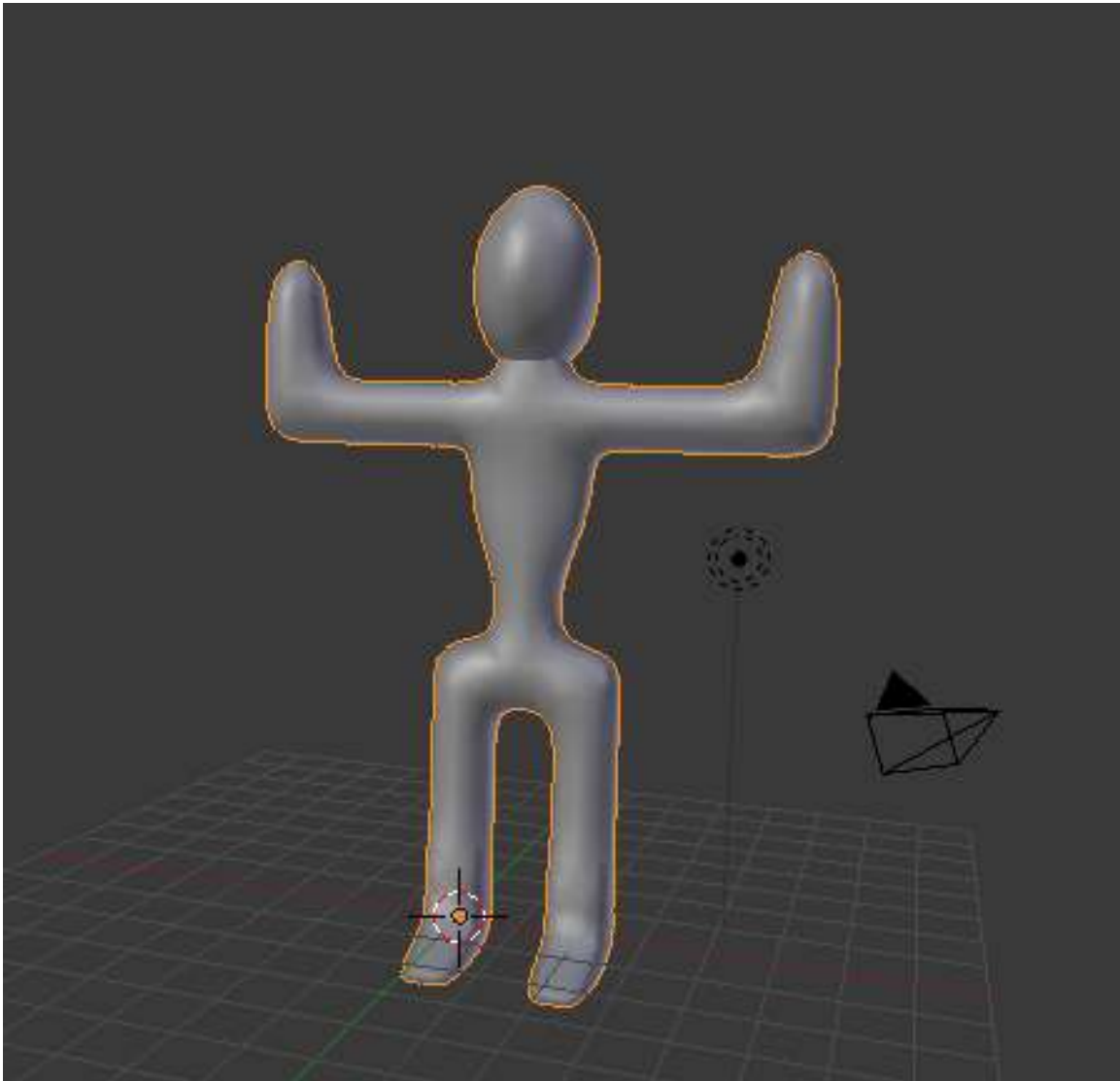


Figure 133

You'll need this simple person later, so remember to save your work!

33. Creating a Simple Hat

In this module, you'll create a hat for your simple person. Along the way, you'll learn how to use the Spin tool and use layers.

33.1. Creating a Generatrix

For future convenience, you'll create the hat as a new object in the scene containing the simple person. If you haven't created the simple person, either go back and do it now or else download the pre-made model from Yosun Chang's website at .

Start by changing layers to layer two, then add the basis for your hat:



Figure 134

1. Make sure you're in Object Mode (so that a new object will be created).
2. Click **[LMB]** on the second little square, this will make the viewport display layer two.
3. Go to orthographic front view by pressing **[NUM1]**.
4. Create a mesh circle at the cursor, by activating the 3D View window, pressing **[SHIFT]** + **[A]** and choosing *Mesh* → *Circle*.

Note:

The new circle will probably look more like a line segment than a circle. If so, it's because you're viewing the circle edge-on.

The new mesh object doesn't actually have to be a circle. You could use any sort of mesh object here because you're about to reshape it into a custom 2D mesh (called a **generatrix**) that describes the profile of your hat. More precisely, the generatrix describes one side of a vertical cross-section through the hat. You'll want your generatrix to have a slope; it should be higher on one side (which will become the top of the crown) than on the other (which will become the brim).

1. The newly-created mesh should be selected. If it isn't, select it by clicking **[RMB]** on it.
2. Press **[TAB]** to edit the mesh.
3. Activate Vertex select mode.
4. Press **[A]** until all vertices are selected.
5. Press **[X]** to erase all vertices.

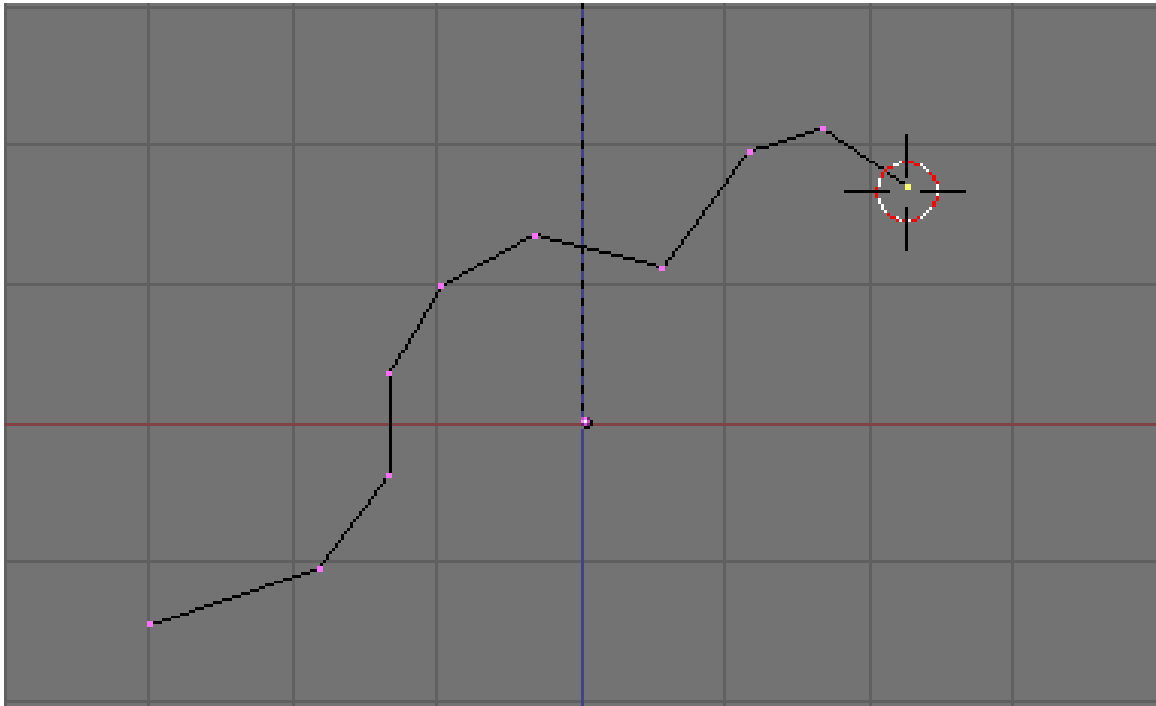


Figure 135

Now draw your generatrix, starting with the brim and sloping upwards toward the top of the crown:

1. Make sure you're still in orthographic front view.
2. Press **[CTRL] + [LMB]** to create the first vertex.
3. Press **[CTRL] + [LMB]** to one side of that vertex to extrude another vertex, connected to the first by an edge.

Keep adding vertices until you're satisfied with the shape of your generatrix. You can always undo using **[CTRL] + [Z]** or go back and adjust the positions of particular vertices using the grab tool.

The Spin control only spins vertices that are selected, so press **[A]** until all vertices in the generatrix are selected. The Spin control spins the mesh around an axis perpendicular to the viewplane. You want to spin around a vertical axis, so press **[NUM7]** to switch to top view.

33.2. Spinning the Hat

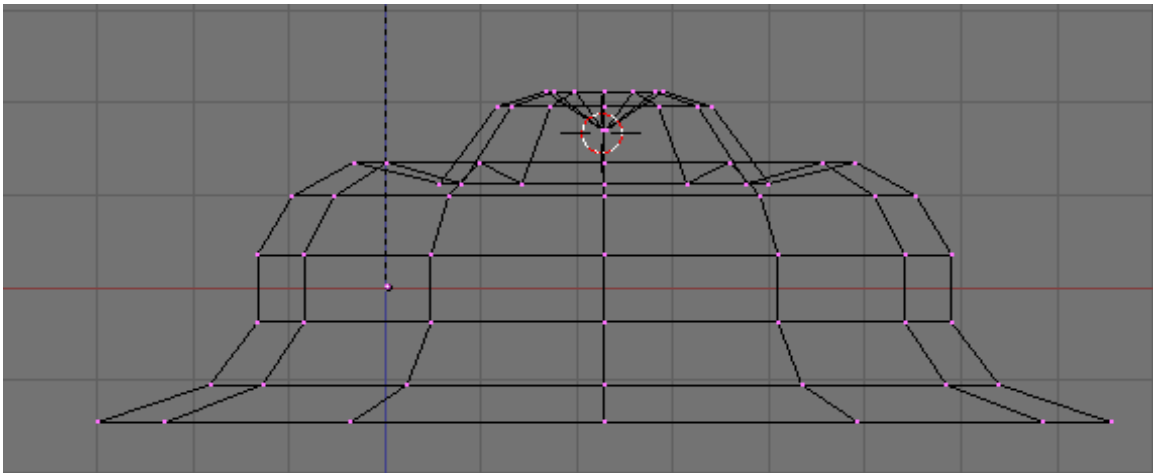


Figure 136 The spun hat, drawn as wireframe in orthographic front view.

Now lets actually spin the hat:

1. Move the cursor to the vertex you want to spin around by pressing **LMB** on it.
2. Press **A** to select all the vertices.
3. Press **ALT+R** to activate the Spin tool.
 - The Spin tool is also available in the Tool Shelf under *Add*

You should now see 90° of a generatrix! To spin your hat all the way round, press **F6** or look in the Operator Panel just below the Tool Shelf. There should be an input slider named *Degrees*, change this value from 90 to 360. There should also be a slider called *Steps*, increase the value from 9 to 15.

Noob note: If your hat has a large hole in the center, check underneath the "degrees" slider. There will be a header that reads "center". Underneath the header there will be three sliders, the first labeled "X", the second "Y" and the third "Z". set these all to zero, by either adjusting them by clicking the arrows on the left and right, or by clicking the middle of the slider itself, and entering "0".

Remember that if you spin an object 360° there will be a double row of vertices at the row of vertices you spun. To fix this, press **A** to select all vertices, press **W** and select *Remove Doubles*. Note that this will only work in vertex select mode.

You may also want to merge the vertices at the top of the hat. Do this by selecting all the vertices at the top with **C** and pressing **ALT** + **M** → *At Center*. You may have to do this twice as some vertices might be beneath each other.

You have more than one 3D View window, so Blender is asking which window to perform the spin in. Click **LMB** on the window that is showing top view.

33.3. Smoothing Your Hat

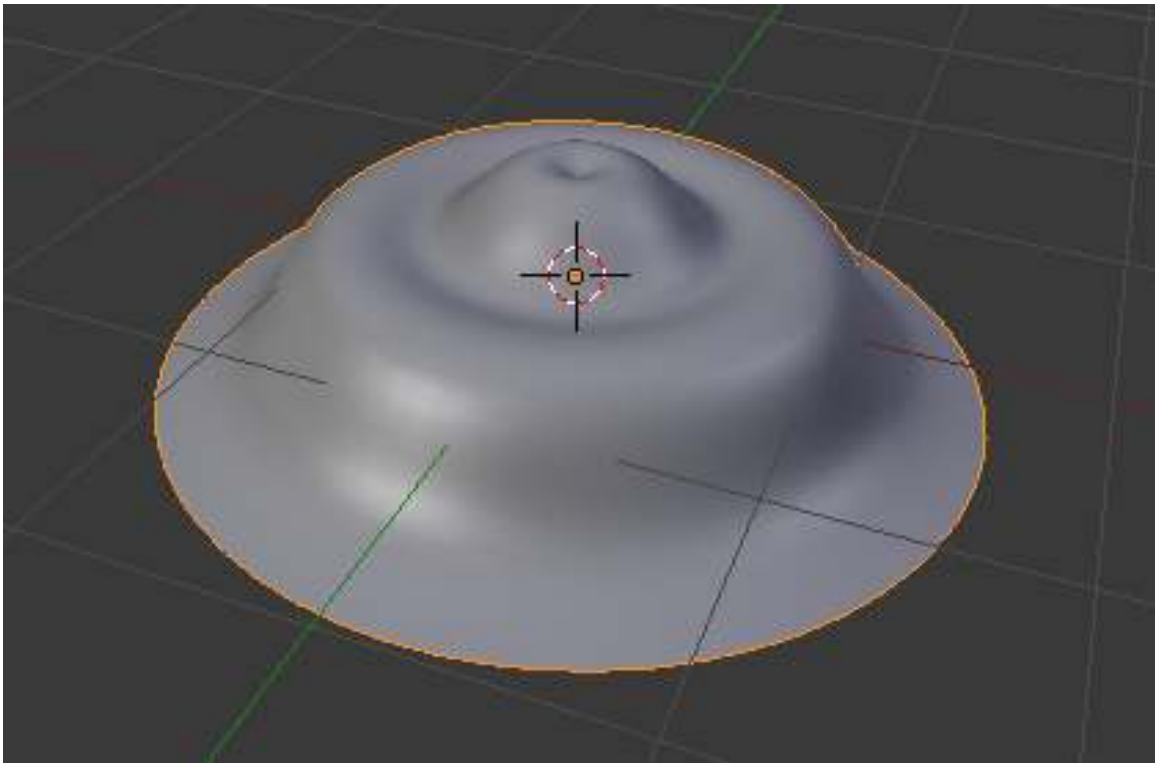


Figure 137 The finished product!

You'll probably have noticed that normal hats aren't usually as faceted as yours! To change this, first change the shading to *Smooth* (available on the Tool Shelf).

Next, add a Subsurf modifier to the hat and set the subdivisions to two, as you did in the "Detailing Your Simple Person 1" module .

1. Click on the modifiers tab (wrench icon) in a Properties window.
2. *Add Modifier* → *Subdivision Surface*.

Save your work. You'll need this scene for the next module.

33.4. Additional Resources

w:Surface of revolution¹

- http://www.youtube.com/watch?v=-mPtxa_MEPA² Ira Krakow's Hat Creation Video Tutorial, based on this page.

1 <http://en.wikipedia.org/wiki/Surface%20of%20revolution>

2 http://www.youtube.com/watch?v=-mPtxa_MEPA

34. Putting Hat on Person

Once you're satisfied with the shapes of individual objects, you'll want to combine them into a coherent scene. You do this in Object Mode.

In this module, you'll learn how to use move objects to and from layers. You'll also learn how to rename and parent objects, and you'll get an introduction to Outliner Windows.

You'll need the person-and-hat scene from the previous module. If you haven't done it, either go back and do it now or else download the pre-made model from Yosun Chang's website at [.](#)

34.1. Adjusting an Object's Median Point

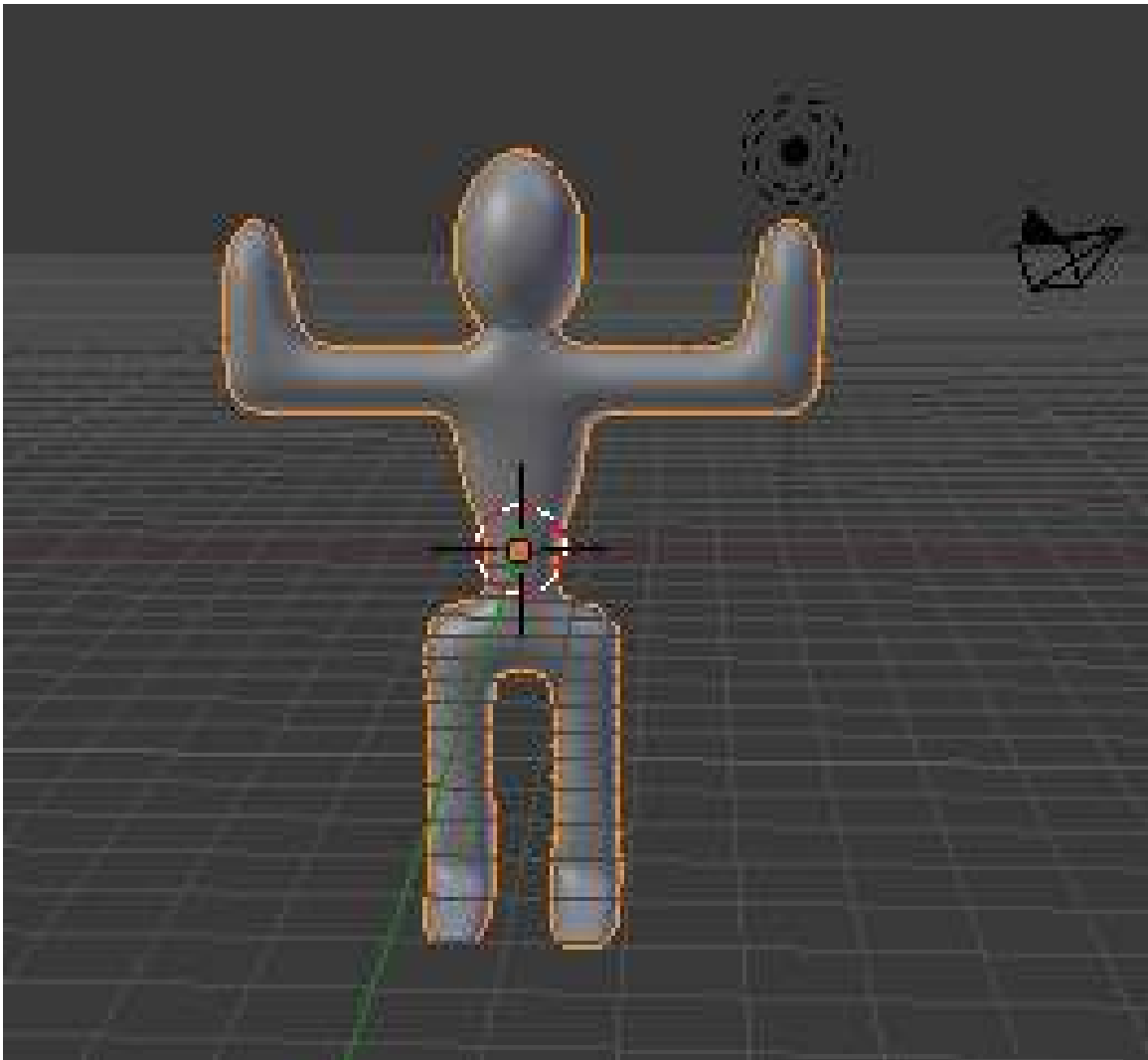


Figure 138 The person that you (yeah you!) made with the origin in his geometric center.

1. Load the person-and-hat scene.
2. Make sure Blender is in Object Mode.
3. Switch to Layer 2, select the hat and press **M**. A dialog box will pop up for you to choose which layer to move it to. Either press **1** (the number on top of the keyboard, **not** the numberpad) or select the first box in the popup.
4. Select the person you made earlier.

Just as you did in Edit Mode, you can specify the pivot for rotating and scaling objects in Object Mode. If you just finished the previous module, the pivot is probably set to "3D Cursor". If so, change it back to "Median Point".

In Edit Mode, the "Median Point" for pivoting is the geometric center of all selected vertices, edges, or faces. In Object Mode, however, it's the origin of the selected object's local coordinates, indicated by an orange dot. In other words, the origin might lie far from the object's geometric center.

You can use buttons in the Tools Shelf to reunify an object's origin with its geometric center:

1. With Blender in Object Mode, click **[LMB]** on the "Origin" in the Tool Shelf and select *Origin To Geometry* to move the selected object's origin to its geometric center (without changing the object's appearance).

Note:

When more than one object is selected, Blender uses the average of their median points as the median point for pivoting.

This can be useful when you want a better picture of your object. With the origin set to the person's geometric center, you can now snap the object with **[SHIFT+S]** to the 3D cursor. This will let you view more of the model at one time and make for a faster editing workflow.

34.2. Positioning the Hat



Figure 139 Positioning the hat

Once you have the hat properly oriented, move it into position on the person's head. The grab tool enables you to position objects in Object Mode in the same way you positioned vertices, edges, and faces in Edit Mode.

1. Make sure Blender is in Object Mode.
2. Click **RMB** on the hat object to select it.
3. Activate the grab tool by pressing **G**.

As you move the mouse pointer, the hat will move around in the viewport. By default, the movement plane is perpendicular to the view axis, so the hat will move differently depending on which viewpoint you're working in.

Just as in the Edit Mode grab tool, you can:

- Restrict the direction of motion by pressing **X**, **Y**, or **Z**. Press once to move parallel to a global axis, twice to use a local axis. (Press the same key a third time to return to view-plane motion.)
- To restrict motion to the global X-Y plane, lock the global Z by pressing **SHIFT** + **Z**.
- Hold down **CTRL** to restrict motion to discrete steps (typically one Blender unit).
- Hold down **SHIFT** to get finer control over the motion.
- Click **LMB** or press **ENTER** to finalize the position and exit the tool.
- Click **RMB** or press **ESC** to return the object to its previous position and exit.

Use two different orthographic views to position the hat on the person's head. You will probably want to scale the hat to make it fit the person's head better. When you are doing this along the X or Y axis, make the changes symmetrical by specifying the axes you want scaling to be constrained to. This option is available in the Operator panel (just below the Tool Shelf) and also by pressing F6.

34.3. Parenting the Hat to the Person

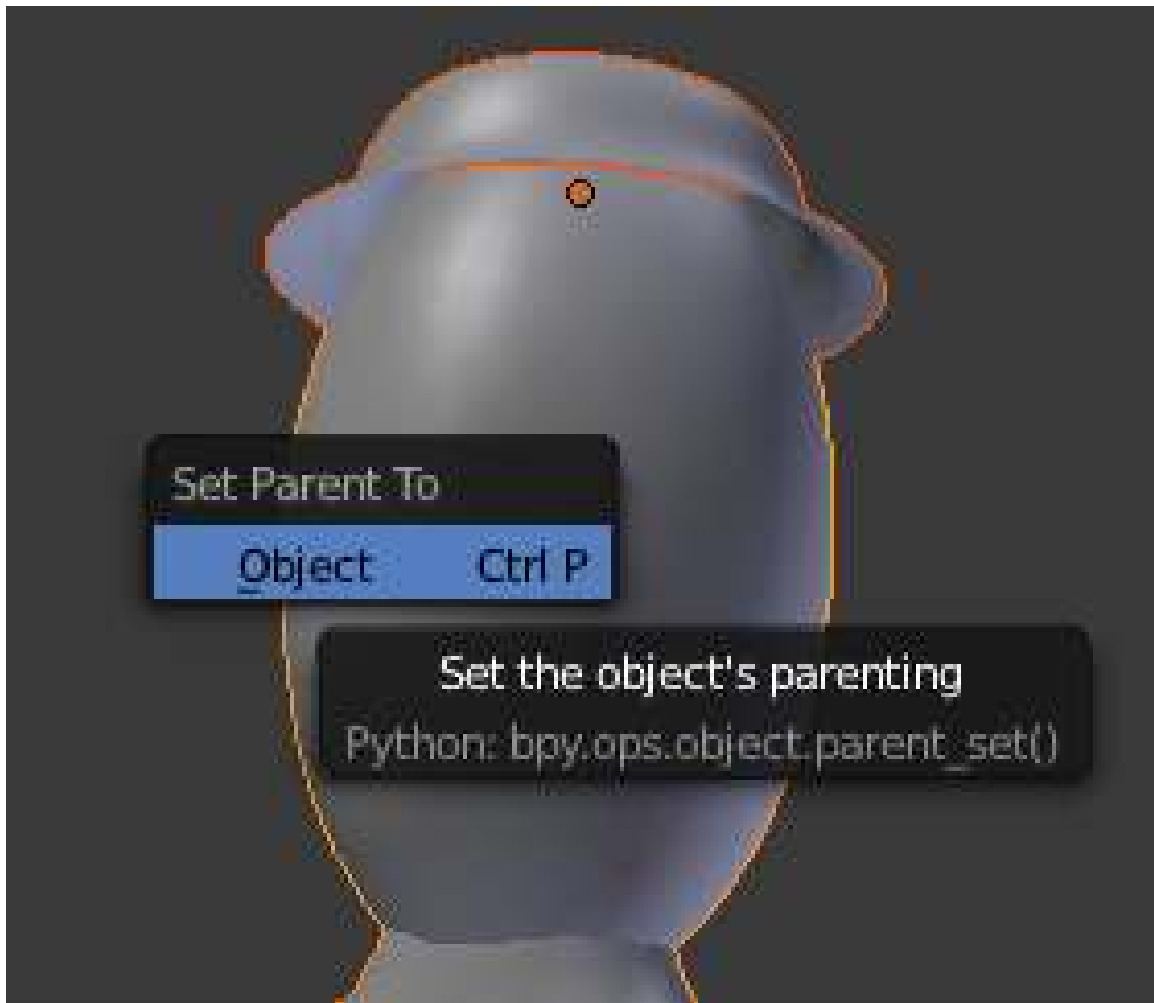


Figure 140 The parenting menu.

Once you have the hat properly sized and positioned on the person's head, you'll want it to stay there. In order to maintain such a cozy relationship between two objects, you'd have to remember to select them both before rotating, moving, or scaling. A drastic solution might be to join them into a single object using **CTRL** + **J**.

A better compromise is to **Parent** the hat to the person. Parenting creates a relationship between two objects, such that certain changes to one object (called the **Parent object**) automatically affect the other (called the **Child object**). Changes to the child, however, do not affect the parent.

Note that an object can have many children, but only one parent.

Since the person is bigger than the hat, it's logical to parent the hat to the person (meaning: parent = person, child = hat) instead of vice versa.

1. Make sure Blender is in Object Mode.
2. Click **RMB** on the hat object to select it.

3. Click **[SHIFT]** + **[RMB]** on the person object.
Both the person and the hat should now be selected. The order of selection is important here.
4. Press **[CTRL]** + **[P]** to parent the hat to the person.
5. Select *Object*. The most recently selected object becomes the parent of all other selected objects.

Note:

Pressing **[P]** in the object mode instead of **[CTRL]** + **[P]** will start the Blender game engine. To stop the game engine, press **[ESC]**.

Now when you move the hat you will see a line from the hat to the person, indicating that the person is the hat's parent. And if you move the person, the hat will move with it.

You may get an error saying something like *Loop to Parents*, fix this by clearing all previous parents with **[ALT]** + **[P]**.

34.4. Renaming Objects

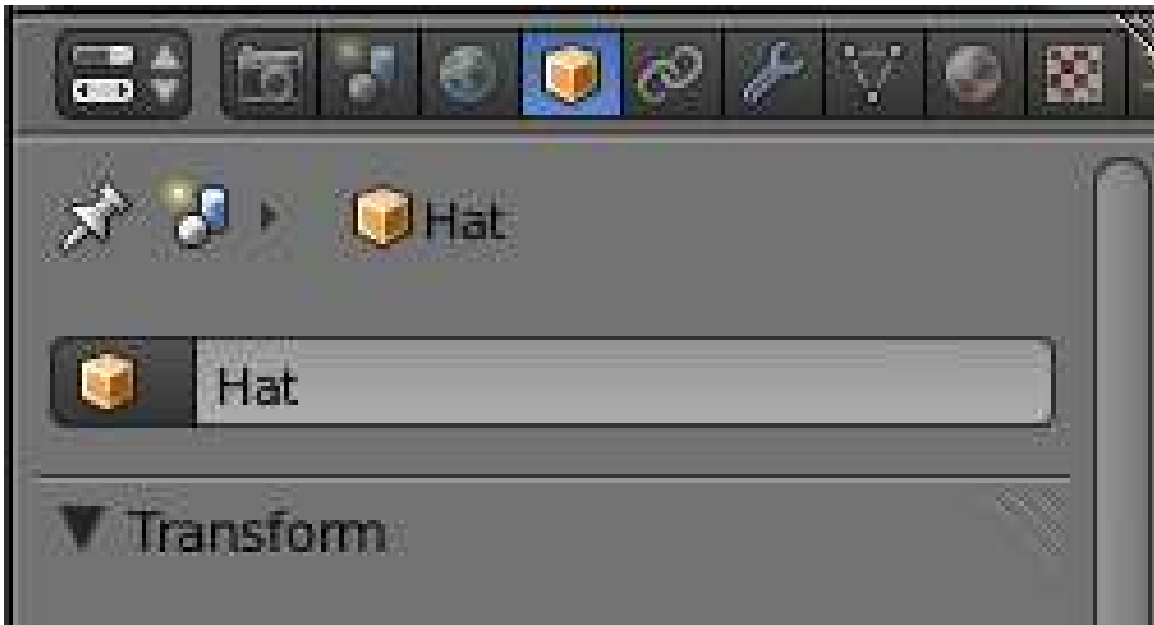


Figure 141 The renaming dialog

When you have multiple objects in a scene, it helps to give each one a name.

Click on the *Objects* tab in the Properties panel (the one with a box icon).

1. Now select the hat by clicking **[RMB]** on it.
2. At the very top of the tab you should see a dialog box with the name of your object
 - The hat's name might be something like "Circle" depending on which mesh primitive you first built the hat from.
3. Click **[LMB]** on the dialog box and type in a more descriptive name like "Hat".

You have now changed the name of the hat's datablock. This name change will be reflected in the Outliner, which we will look at shortly.

Now select your person by clicking **RMB** on it and repeat the process, changing the name to something like "Person".

34.5. Outliner Windows

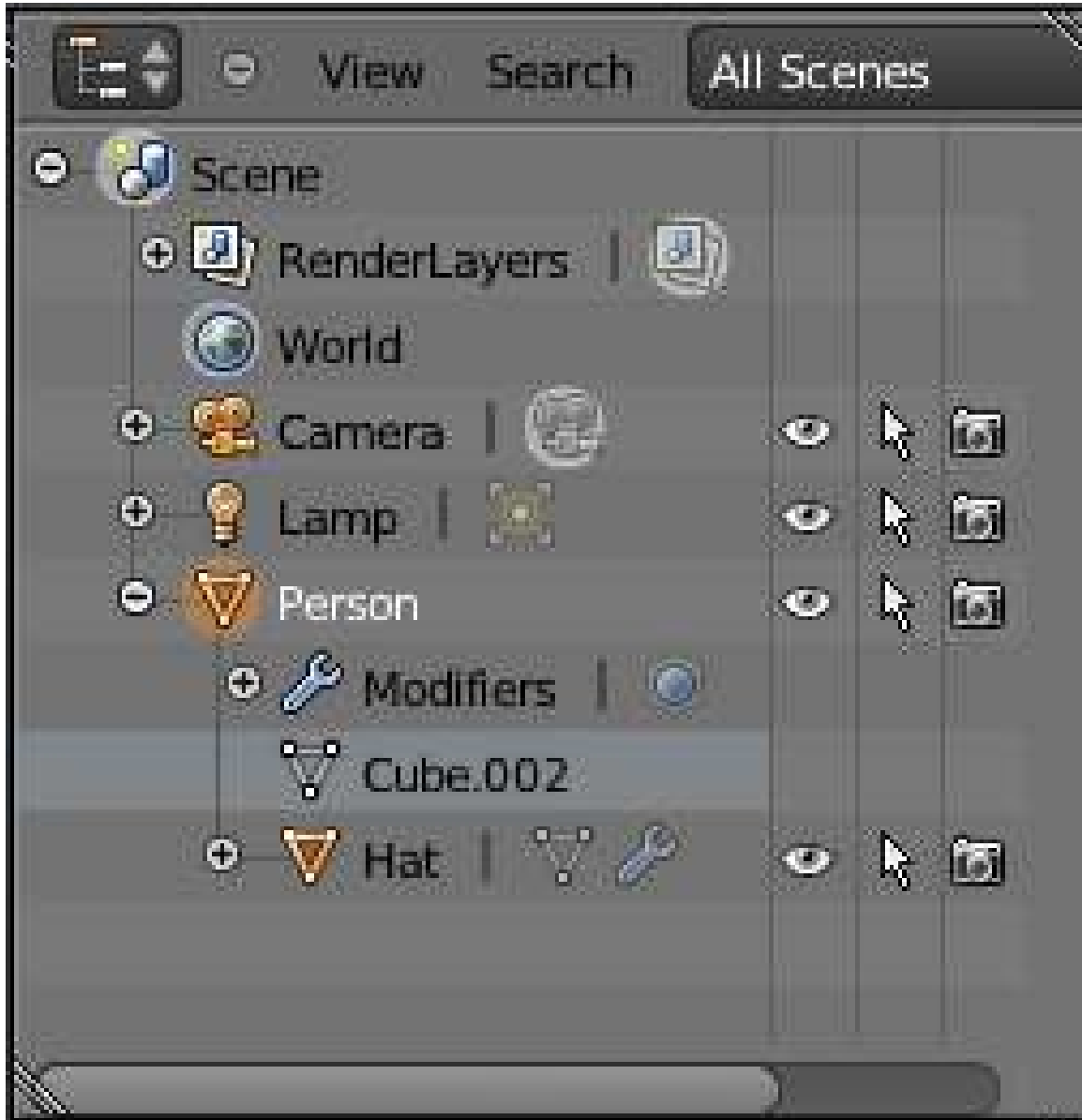


Figure 142 The Outliner window.

Once you give objects names, it helps to have a way to find objects by their name and parent. This is exactly what the Outliner is for and it comes in very handy when you are working with a large scene.

The Outliner is usually just above the Properties panel, you may want to pull it down a bit to see it more clearly.

You'll notice that all the objects in your scene (Person, camera etc) are listed and that you can select these objects by clicking **LMB** on them. And if you click **RMB** on an object, a menu will pop up with options like *Select*, *Deselect*, *Delete* etc. If you select the Person and then click the "+" sign to its left, you will see that the Hat is listed below the person. This is because Blender lists all children objects beneath their parents.

On the right of each object there are a series of icons which represent the state of the object. For example, The eye icon means that your object is visible in the 3D viewport. You can turn off its visibility by clicking **LMB** on the eye, which will turn grey; click again on the eye to make it visible. If you hover the mouse over the icons a text box will pop up with a description of what that particular icon does.

34.6. Good on ya' mate!

Congratulations!! You have now finished your simple character. Pat yourself on the back, and have a celebratory coffee! (Or pop!)

34.7. Additional Resources

- Outliner at
- The Blender Manual page on Gestures at <http://wiki.blender.org/index.php/Doc:Manual/Gestures>
- The Blender Manual page on Groups and Parenting at <http://wiki.blender.org/index.php/Doc:Manual/GroupsandParenting>

35. Materials and Textures

In 3D graphics, materials and textures are nearly as important as shapes. Scenes would be boring if all the objects were gray.

The **material system** in Blender allows you to model a wide variety of materials and how they interact with light. The next few modules will introduce the available options.

35.1. Material versus Texture

A **material** defines the optical properties of an object: its color and whether it is dull or shiny. A **texture** is a pattern that breaks up the uniform appearance of the material. Very few objects in the real world have completely uniform surfaces. Instead most of them have patterning or variation in color: consider the grain in a piece of wood or the pile in a carpet or the mortar in a brick wall.

Blender allows textures to influence materials in various ways, such as altering their colors. Multiple textures can interact with each other to produce interesting effects.

It is important though, to set your material first in Blender before the texture can be applied. This is compulsory in Blender. You must set your material property first before you can add a texture. But you can't do it the other way round. Just simple as that!

35.2. Other Material Settings

Additional settings you can specify for a material include **shaders**, **ray-tracing** and **halo**.

Shaders determine how the appearance of a material varies with the angle of the light: **diffuse** shaders give a non-shiny look, while **specular** shaders give a mirror-like finish. Blender's material settings always involve both kinds of shaders, but you can adjust a material's diffuse and specular colours separately to control their respective effects; if you set the specular colour to black, the surface will no longer produce reflections.

Ray-tracing is a technique for modeling the physical path of light through the scene. It is capable of producing exquisite reflection and refraction effects, including different degrees of reflectivity, translucency and transparency, and representing materials with different indexes of refraction. Blender provides two separate groups of ray-tracing settings, one for reflection of light and the other for its transmission through the material. You can control these settings on a per-material basis.

Halo rendering means an object no longer looks like solid matter, instead it appears to be made of bits of light. This can be used for real-world effects like fire, smoke and plasma, or to create fantasy effects with no connection to reality.

Note that reflections produced by ray-tracing are separate from that produced by the specular shader: the former are controlled by the material's mirror colour, while the latter is controlled by its specular colour.

Reflection is done two different ways because, while ray-tracing produces the most realistic renders, it is also very CPU-intensive. It is therefore best to apply the ray-tracing effects when you're completely done with your modelling to help reduce high CPU usage. Enough practice with the ray-tracing can also help you get stunning effects with just few clicks without you having to do much trial and error. Do well to dedicate at least some hours of your time in experimenting with it, so that in a future real production time, you spare yourself of all that hassle.

35.3. Types of Textures

When you create a texture in Blender, you will see a popup menu listing a whole lot of different types for the texture. The **image texture** type lets you use a scanned image to texture your object: for example, you can scan an actual piece of metal and use that to give your object a realistic metallic appearance, or use a photograph of an actual brick wall to texture the wall of a building model, and so on.

The other texture types are called **procedural**, which means the textures are generated according to algorithms built into Blender itself. These can be useful for simulating various effects when you don't have an image of the real material handy; they can also be applied to augment the appearance in various ways. For example:

- using a “cloud” texture to “dirty-up” a material
- or
- using one texture as a stencil to create an amalgam of two other textures.

35.4. Additional Resources

w:Diffuse reflection¹ w:Ray tracing (graphics)² w:Specular reflection³ w:Texture mapping⁴

1 <http://en.wikipedia.org/wiki/Diffuse%20reflection>

2 <http://en.wikipedia.org/wiki/Ray%20tracing%20%28graphics%29>

3 <http://en.wikipedia.org/wiki/Specular%20reflection>

4 <http://en.wikipedia.org/wiki/Texture%20mapping>

36. Quickie Material

In this module, you will create a new material called "green ooze". Along the way, you will learn how to alter the diffuse, specular, and mirror colors of a material.

36.1. Your First Material



Figure 143 **Figure 1:** The Materials tab in the Properties panel.

The cube in the default scene (which you get from *File* → *Load Factory Settings*) has a simple grey color. Now click on the Materials tab



Figure 144

in the Properties panel.


The materials tab contains various menus, but for now you only need diffuse, specular and mirror. The material is named and linked in the panel above the preview window. (**Linking** is a feature that allows materials to be shared between multiple objects (or datablocks). Changing a material affects the appearance of everything it is linked to.)


The first row of the window above the preview window indicates that:


- there is one material assigned to this object and its name is "Material".

The second row of controls indicates that:

- The current selected material's name is "Material".
- This material will only be saved if it's in use.
- It is not a "Nodes" material.
- Instead of being linked directly to an object, the current material is linked to a datablock.

To rename the material, click  on the name and enter the name you want.

To unlink the material, click  on the **X** button to the right of the material name ("Material"). This deletes the link to the datablock, removing the material from the mesh. As a side-effect, most of the panels in the Material tab disappear. You will see in a moment, however, that the material still exists. It hasn't been deleted; it is simply no longer in use.

At this point, you could click  the "New" button to create a new material, but instead we are going to reapply the old material:

1. Click  on the



Figure 145

button to the left of the "New" button.

2. You'll see a nifty drop-down list containing all materials you've created so far. Choose *0 Material*.

Materials whose names are preceded by "0" in this list are not in use. By default, Blender doesn't save such materials when it saves the scene. Thus, you can delete a material from the list by saving the scene and then reopening it. You can override this behavior by toggling the "F" button "on" for unused materials you want saved.

Your materials will be much easier to find and manage if you give them brief, descriptive names you can recognize at a glance. Change this one's name to "Green Ooze". In addition, naming of your materials and other objects in your scene is useful when such components of your scene will be appended in another scene of a different Blender file. Naming your materials and other stuffs

on scene will enable you choose the right objects and materials you need whenever you wish to append just a portion of a whole bunch of work you did. For instance, you're working on a new blender project, but felt the material you used in this Blender file is worth it. Going through the pain of creating a new material (of course you guessed in the initial one in getting the right material appearance), you just append the material to your new work. Pretty simple! Make naming a habit, as it's much used in a production environment.

36.2. Specifying Colors

Since changing the name of the material doesn't change its properties, you still have some work to do.

Simple materials are specified by three colors: diffuse, specular and mirror. White rectangles in their own panel in the Material tab allow you to access these three colors one at a time. Diffuse color is the basic underlying color of the material, rendered by the diffuse shader. Specular color is for highlights (small bright spots on a shiny surface) as rendered by the specular shader. Mirror color is for true reflections rendered using ray-tracing.


- You can specify a color by its red, green, and blue components. By mixing different amounts of red ("R"), green ("G"), and blue ("B"), you can get any color that a computer monitor can display. 0.000 means none of the component, and 1.000 means the maximum amount of the component. For instance, R=1.000, G=1.000, B=1.000 produces white.

If you click on the white rectangle under the diffuse panel you can use the RGB color wheel or the color sliders to change the values. This is the most intuitive way. The window that pops out will look like this and will include the following (Figure 2):




Figure 146 **Figure 2:** colorwheel

1. A color wheel to change the color as you want.
 2. Three color sliders that will change if you change the color in the colorwheel. You can also change the values with the sliders.
 3. A slider that controls the intensity of the color.
 4. A pipette capable of sampling colors from any Blender window or render window.
 5. Buttons that can change it to "HSV" or "HEX" mode.
- Alternatively you can specify hue, saturation and value components by clicking LMB on the "HSV" button and pushing the sliders around accordingly.
 - You can also press the last button and enter the hexadecimal (or HEX) code.

If you want to get rid of the window just click  anywhere else.

The most used method of creating a color of your own is using the color wheel, but because we want to be sure you will get the exact same color as us we will use the sliders. Use the above methods to set the diffuse color to R=0.149, G=1.000, B=0.446 (or use the HEX code: 6CFFB2). If you look in the "Preview" panel, you will see that the material is now bright green.

Most real-life materials (other than metals) don't alter the color of specular light. For this reason, Specular and Mirror are usually left at their default values (white). For green ooze, however, you'll disregard this rule-of-thumb:

1. Click  the sample rectangle below the Specular window.
2. Use the color selection dialog to adjust the specular color and watch the Preview panel to see how this color affects the sample sphere's highlight.
3. Set the specular color to R=0.640, G=0.990, B=0.566 (or use the HEX code: D1FEC6).

With these values for Color and Specular, you should be able to get a good ooze later on. The Preview, Diffuse and Specular panel should now look like this:



Figure 147

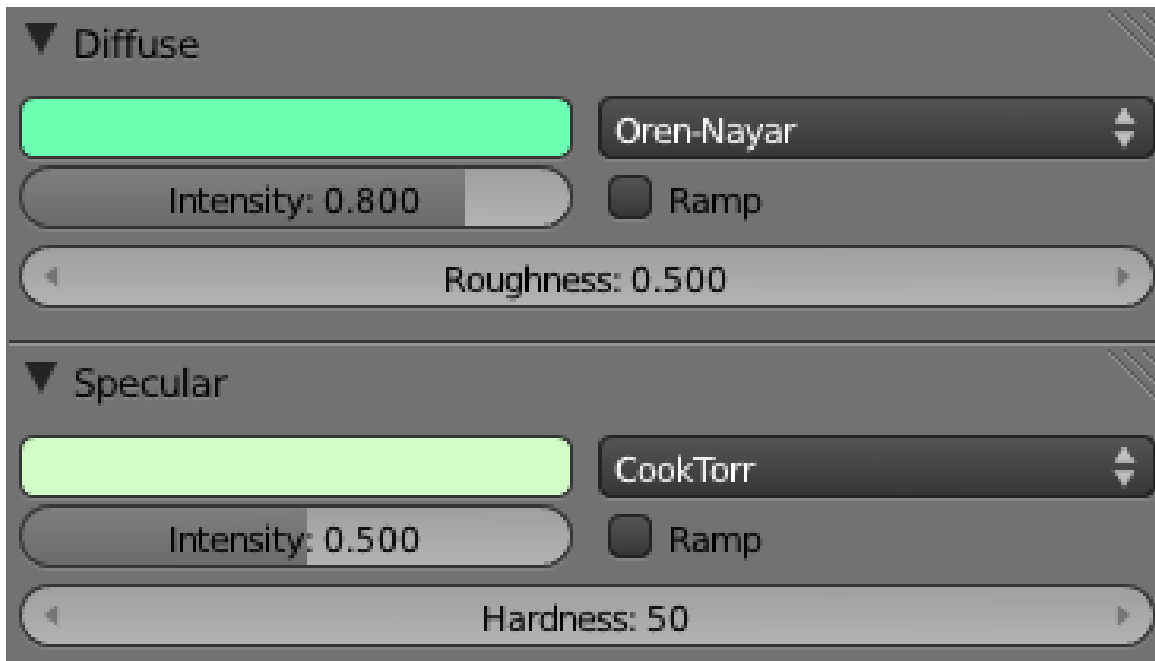


Figure 148

As you can see, there are many other material buttons. Many of these will be explained in later modules. Suggestions for creating specific materials may be found in the "Every Material Known to Man" module .

Save this scene before proceeding. You will need it for the "Quickie Texture" module , in which you will perfect your ooze.

36.3. Additional Resources

- w:HSL and HSV¹
- w:RGB color model²
- How to Assign a Different Material to Each Face of a Cube at www.youtube.com/watch?v=hCYViRjFf5w

This tutorial was created using Blender v2.59

¹ <http://en.wikipedia.org/wiki/HSL%20and%20HSV>

² <http://en.wikipedia.org/wiki/RGB%20color%20model>

37. Multiple Materials Per Object

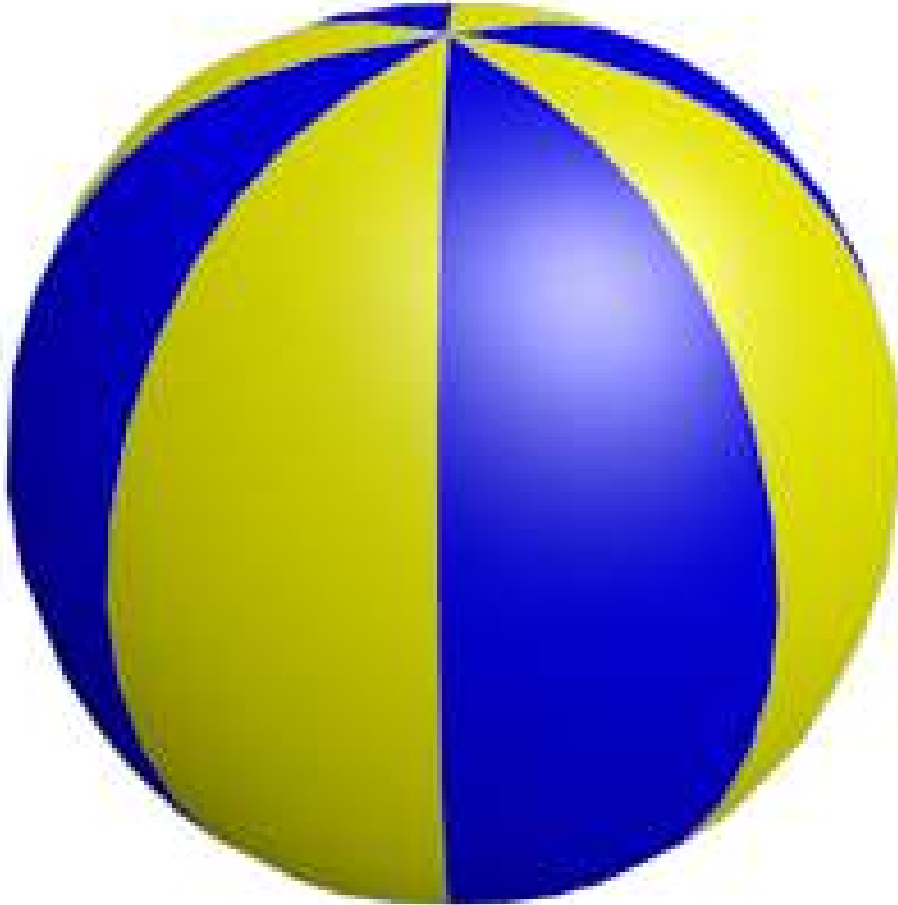


Figure 149

In this module, you'll create a beach ball with two alternating colours. Along the way, you'll learn how to apply multiple materials to a single object.

Many real-life objects have parts which are different colours, or are even made of different materials. One way to model such objects is to make each part a separate Blender object. However, Blender also allows you to assign different materials to parts of a single object.

37.1. Set the scene

Begin by opening blender and removing the default cube. Then select the lamp and change its type from 'Spot' to 'Hemi' in the Light settings in the the properties window (object data button), this has the advantage of it giving more light.



Figure 150 The options for changing lamp type in the properties window

Now create a mesh for the beach ball:

1. With the 3D View window active, press $\text{[SHIFT]} + \text{[A]}$) and choose *Add* → *Mesh* → *UV Sphere*.
2. Set the controls under "Add UV Sphere" at the bottom of the left panel to 8 segments and 4 rings.

The initial result will be crude, but meshes with fewer vertices are easier to edit.

To make the mesh more round, go to the modifiers context in the properties window (wrench icon) and add a subsurf modifier. Set the values in the 'View' and 'Render' settings to **2**. And to get rid of that blocky look, go to the Tool shelf and set the shading to *Smooth*.

The ball is now round, but a bit prolate. To make it more spherical, scale it by about 1.1 along the X and Y axes. To select the X-Y plane, you select '*not Z*', by using the key combination $\text{[SHIFT]} + \text{[Z]}$. The complete sequence is, then, [S] , $\text{[SHIFT]} + \text{[Z]}$, [1.1] .

37.2. Colorize Time

Now you're ready to begin adding colors to the object:

1. Press [TAB] to put Blender into Edit mode.
2. Select the "Materials" button



Figure 151

and press New

Several additional panels should appear in the Properties window.

3. In the "Material" panel, set the Diffuse colour to one of your two chosen stripe colours, say, yellow.

At this point, the entire ball is yellow.

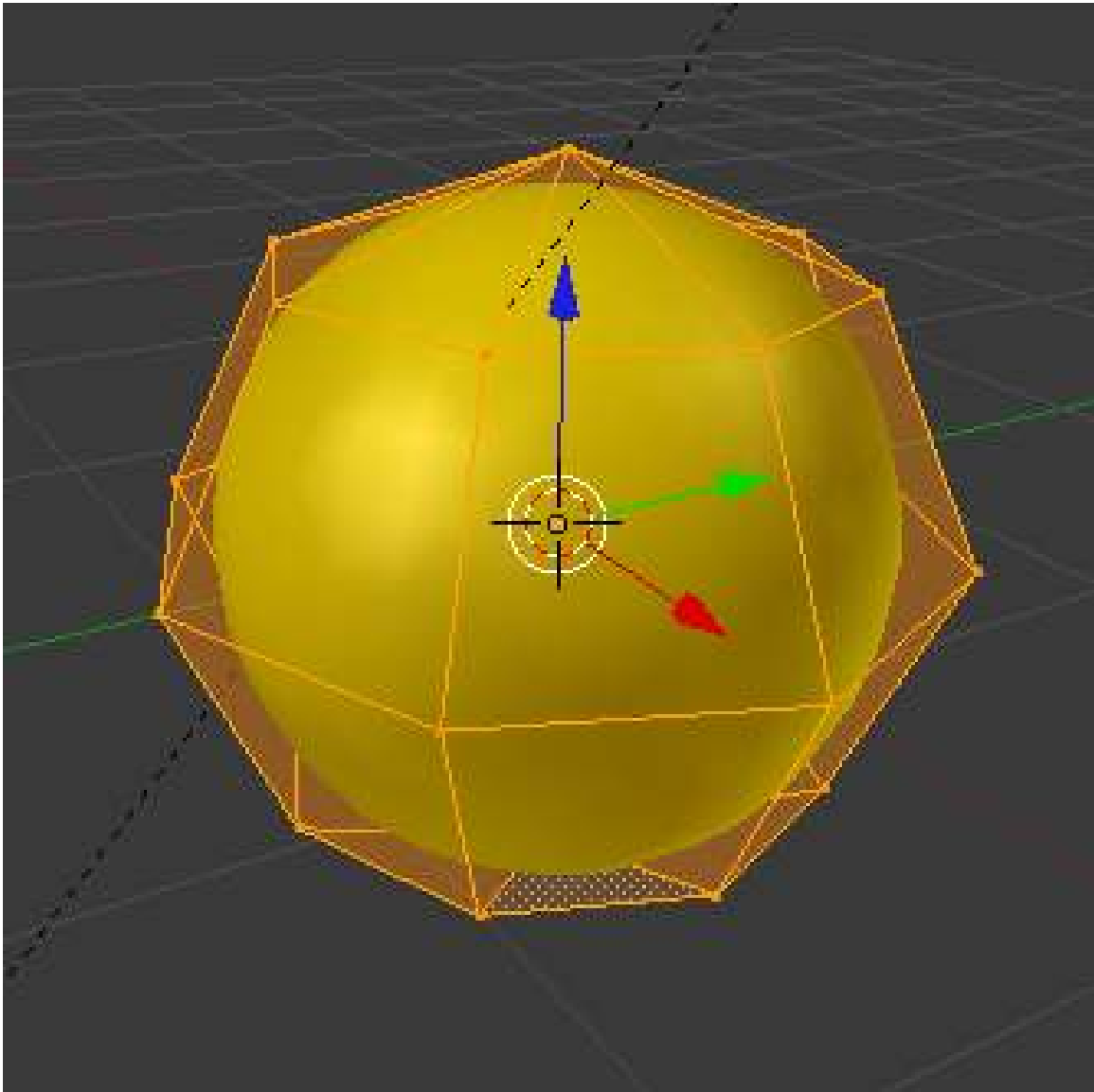


Figure 152

In the "Materials" panel click the "+" button (indicated by the red box in the picture, below) near the top of the panel and to the right of the existing material list. This will create a new blank slot in the materials list. A "+New" button will appear (indicated by the blue box, in the picture below).

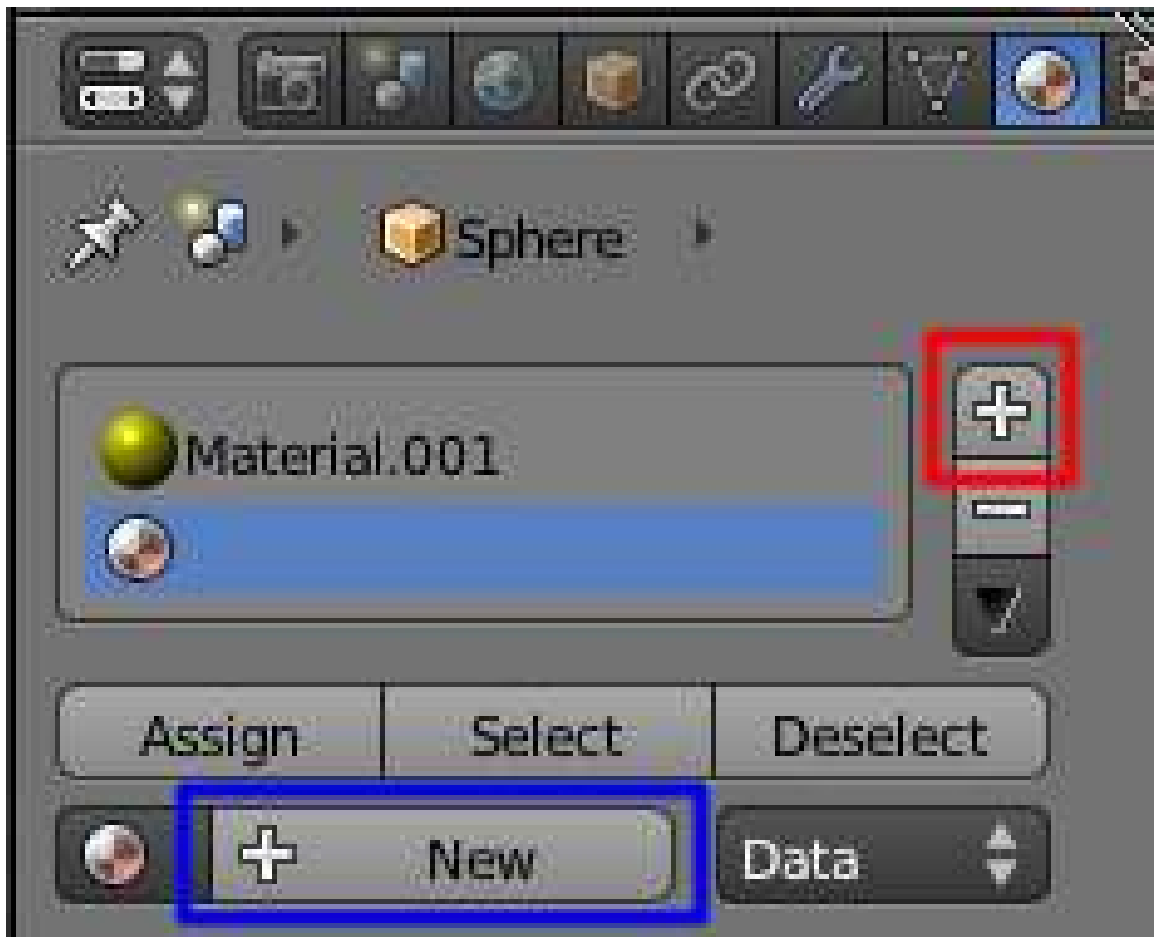


Figure 153

Click the +New button and a new material will appear in the materials list. Ensure it is selected in the list of materials and change the Diffuse color to the second color of your choice (the images in this tutorial use blue). Nothing will happen to the beach ball, yet.

All the vertices should still be selected from before; hit **A** to deselect them. Switch to front view with **NUM1**, and to "Face Select" mode with **CTRL** + **TAB** then **F**. To avoid accidents, make sure you have "Limit selection to visible" enabled. Now select a column of four faces that will make up one stripe of the beachball (using **SHIFT** + **RMB**):

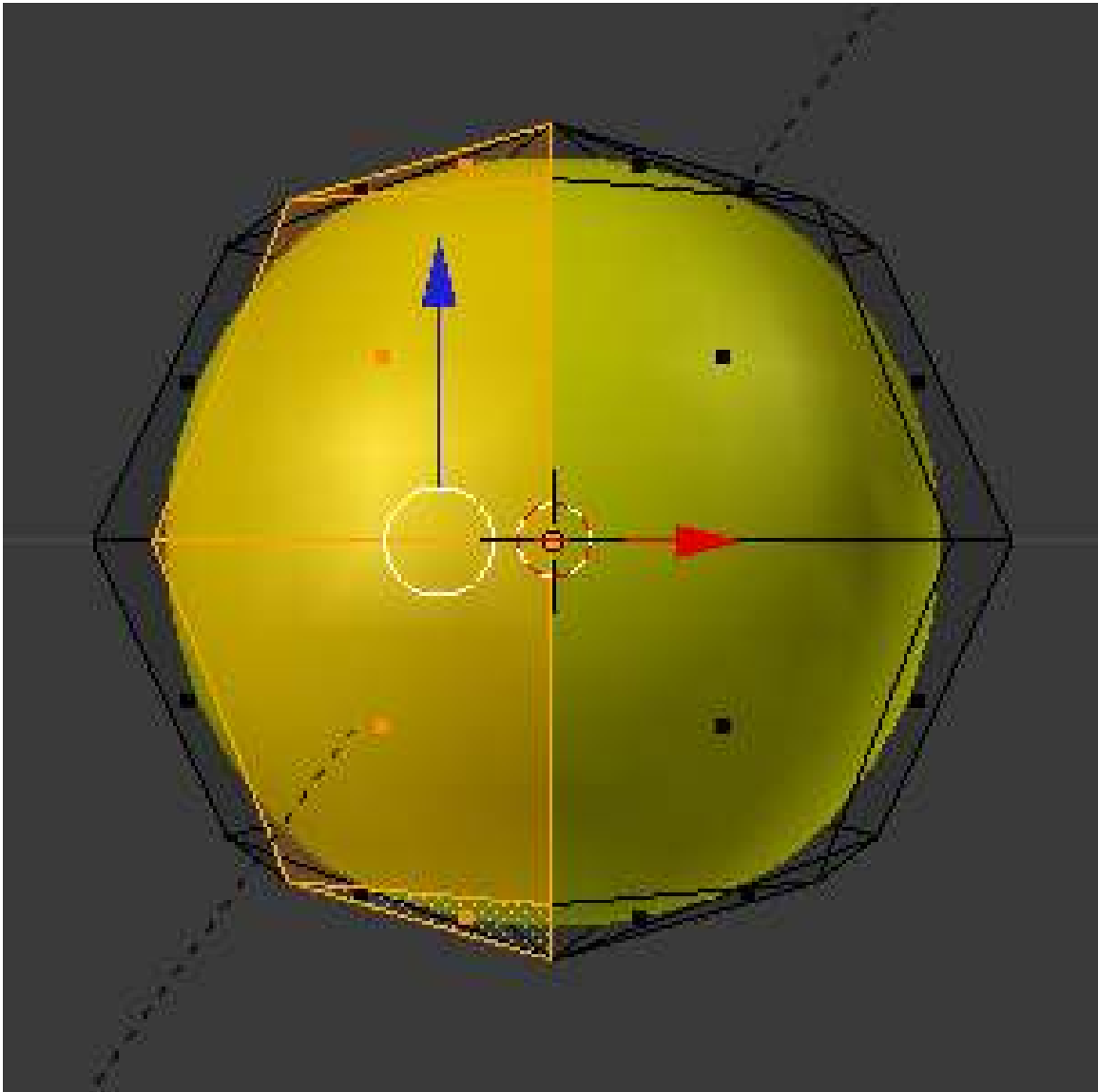


Figure 154

In the "Material" property window, highlight the second chosen material ("Material.001") in the materials list and click the "Assign" button.

Now rotate the view (e.g.) so you can skip past a yellow stripe adjacent to the blue stripe, and select the second column that will become a blue stripe. Work your way around the ball to do this two more times. (Remember we made the sphere with 8 segments; four of these are the yellow stripes, and four are the blue). Now you see why I chose to make the sphere with only 4 rings: more rings would have meant more faces in each stripe, and more clicking to select them.

38. Image Textures

Procedural texturing is very powerful; however, sometimes it is difficult or impossible to generate the desired realism with them. Image texturing is there for you when you need it. To review, the basic idea is to take an outside image and wrap it around your model. You can use any texture, or a seamless one if you want it to repeat to get a tiled effect. The following shows how you create a seamless texture, and then how to apply any texture (seamless or otherwise) to an object.

38.1. Easy Image Texture Creation With Gimp

It is easy to create a seamlessly-tiling texture image with The GIMP¹. Start with the photo you want to use. Crop out any part you don't want. Here's an example random photo of some plants in my garden:



Figure 155

¹ <http://www.gimp.org>

Go to Gimp's "Filters" menu, and find the "Map" submenu. In here you will find the entry "Make Seamless". Select it. That's it:



Figure 156

Just to prove it works, here's a (scaled-down) use of the result as a tiled fill pattern:



Figure 157

38.2. Other Image Texture Editors

- Wood Workshop² A free utility (Requires Operating System: Windows 2000/XP) that generates surprisingly high quality tiling wood texture images. These textures can be exported as standard image files for use within Blender.
- MapZone³ A free utility for Windows (works perfectly in Wine⁴) that generates node based procedural texture maps. Mapzone can export diffuse, normal and alpha texture maps as standard image files. It can also import SVG regions created with Blender's UV mapping tools.

38.3. Mapping an Image Texture to an Object

Now that you have an image, either a seamless one as above, or any image at all, you can apply it as a texture on an object. To do this:

2 http://www.spiralgraphics.biz/ww_overview.htm

3 <http://www.mapzoneeditor.com/>

4 <http://winehq.org/>

1. Make sure you are in the Materials panel of the Properties Window
2. Highlight the material where you want your image to appear
3. Now, switch to the Image panel

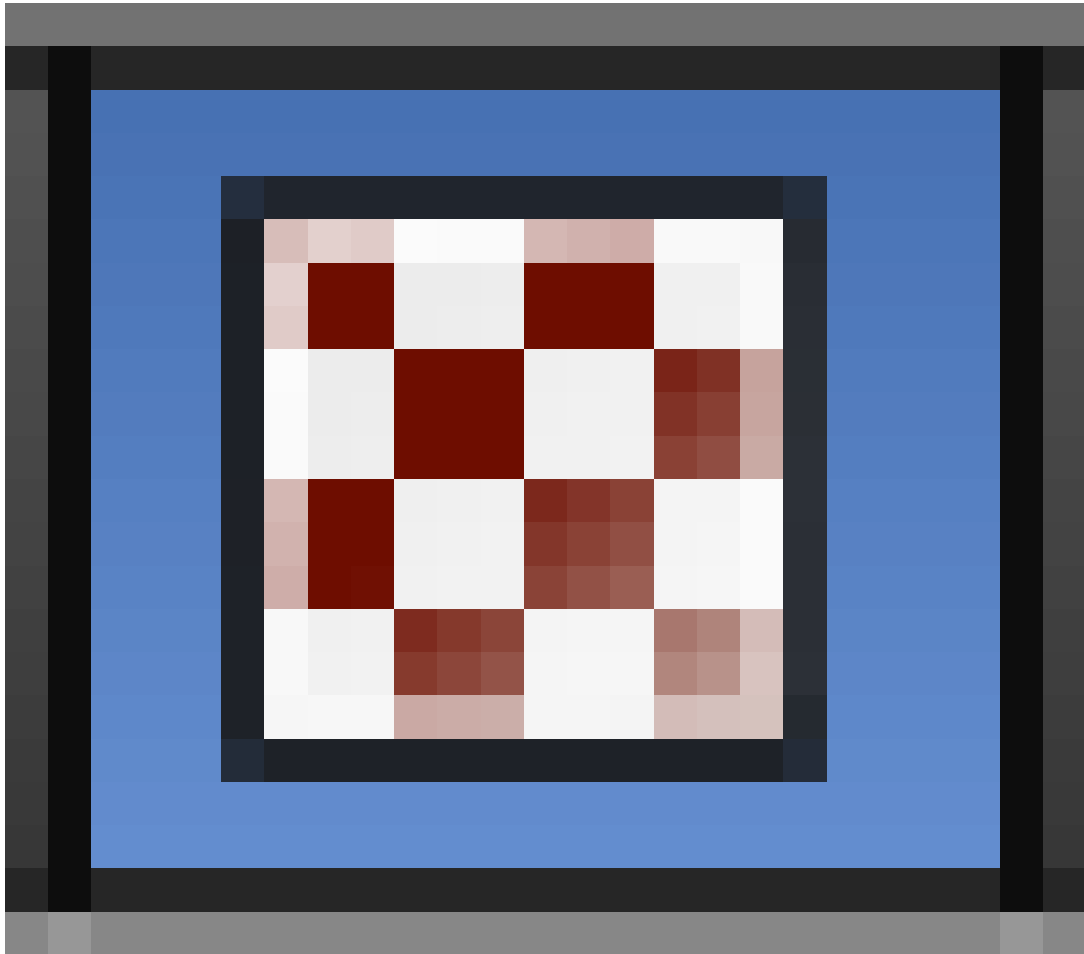


Figure 158

of the Properties Window

4. Click on one of the unnamed textures in the texture list and click on the 'New' button below it
5. Blender will automatically create a 'Clouds' image texture. Click on the Texture Type drop down menu and change the type from 'Clouds' to 'Image or Movie'
6. In the Image window further down, click **Open** and navigate to your image on your computer and Open it. Once loaded, your image will appear tiled in the Preview window.

To see the image on your object, you'll need to Render it using the Render menu or F12

See Using Textures⁵ for more details on applying images as textures, and using them to affect many other surface attributes such as luminosity, reflectivity, translucency, displacement etc.

5 Chapter 72 on page 655

39. Procedural Textures

Procedural Textures

Texturing objects can be broken down into two categories: procedural and image texturing. Procedural texturing makes use of mathematical formulas to generate textures. This is nice because it can be used to make relatively nice looking textures without external images which are very temperamental where you put them. Procedural Textures are all stored in the .blend file. These textures are obviously generated within Blender itself. Image texturing uses images created or captured outside of Blender, either from an image manipulation program such as the Paint.NET, GIMP or Photoshop, or captured on a camera. We have already learned about image texturing, so let's move on to procedural texturing.

Current Procedural Textures

Blender currently supports many procedural textures, including: Clouds, Marble, Stucci, Wood, Magic, Blend, Noise, Musgrave, Voronoi and DistortedNoise.

39.1. A Simple Wood Texture

Normally you have to assign a material for a new object. The default cube already has a material assigned to it. So we can directly proceed and give it a texture. In the Properties window, click on the "Texture" button

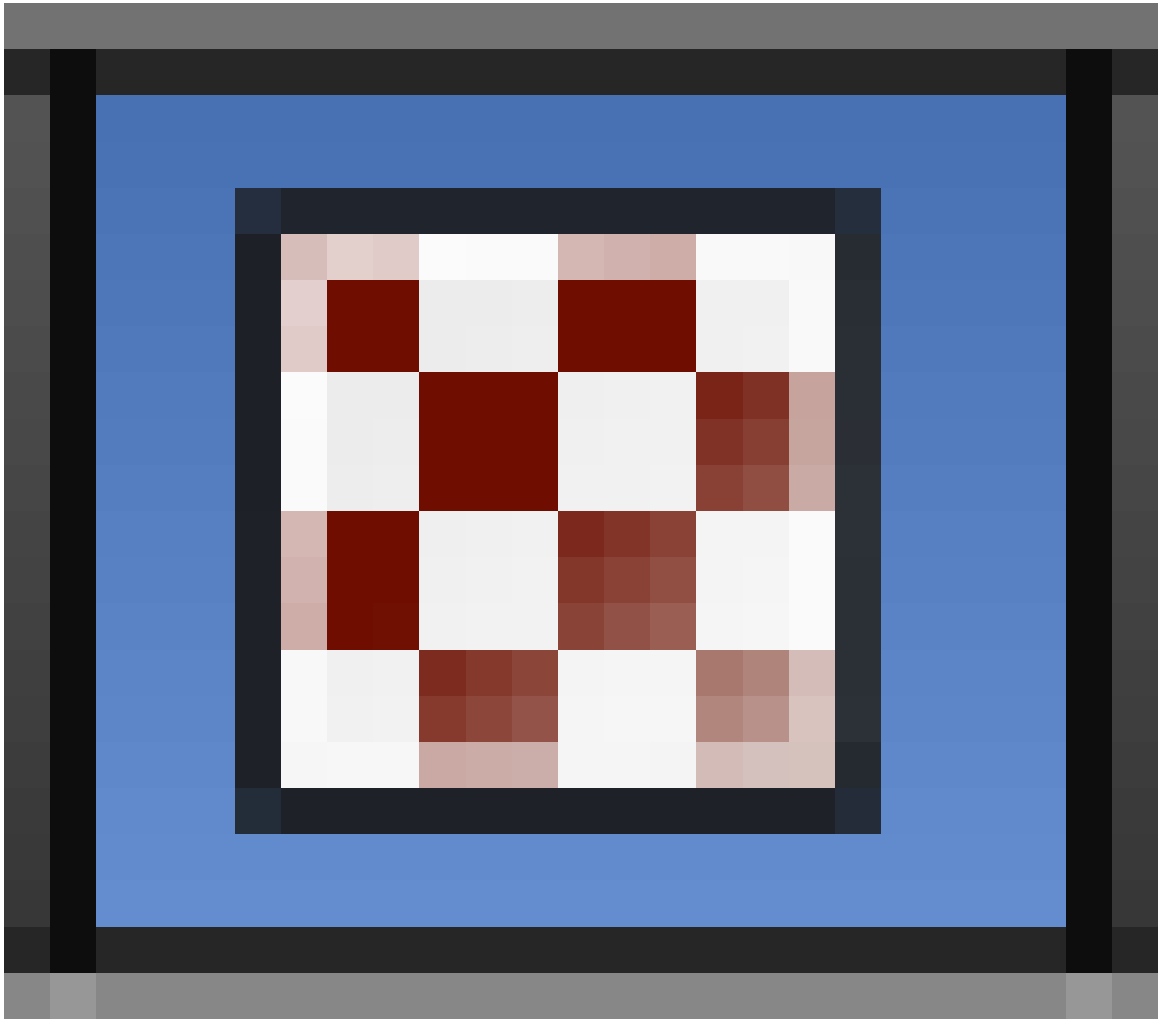


Figure 159

. On a new object you have to create a texture first as well. The default cube already has a texture assigned. Under “Type”, click on the button “None”. In the “Texture Type” popup menu that appears, select “Wood”. The texture sample will show parallel alternating black and white bars that don’t look very woody at all. But never fear.

In the “Wood” properties panel that appears, change the waveform from “Sine” to “Saw”. In the next row of buttons down, change the type from the default “Bands” to “RingNoise”. Increase the NoiseSize to 1.0. Now the texture sample should show something resembling wavy tree-rings. If you hit F12 to render now, you will see these rings covering your cube, except a) the colour is wrong, and b) normal wood patterns aren’t so nearly circular.

To make the pattern more elongated, go to the “Mapping” properties panel, change the Size X value to 2.0 and Y to 0.4. Hit F12 to render again, and the shape of the texture should be looking a lot more woody now.

To fix the colour, go to the “Influence” properties panel further down, click on the colour swatch, and choose a nice brown colour. For a nicer effect, I chose a very light brown. Switch to “Material”, go

to the “Diffuse” properties panel and choose a darker brown color. The result looked very woody indeed:

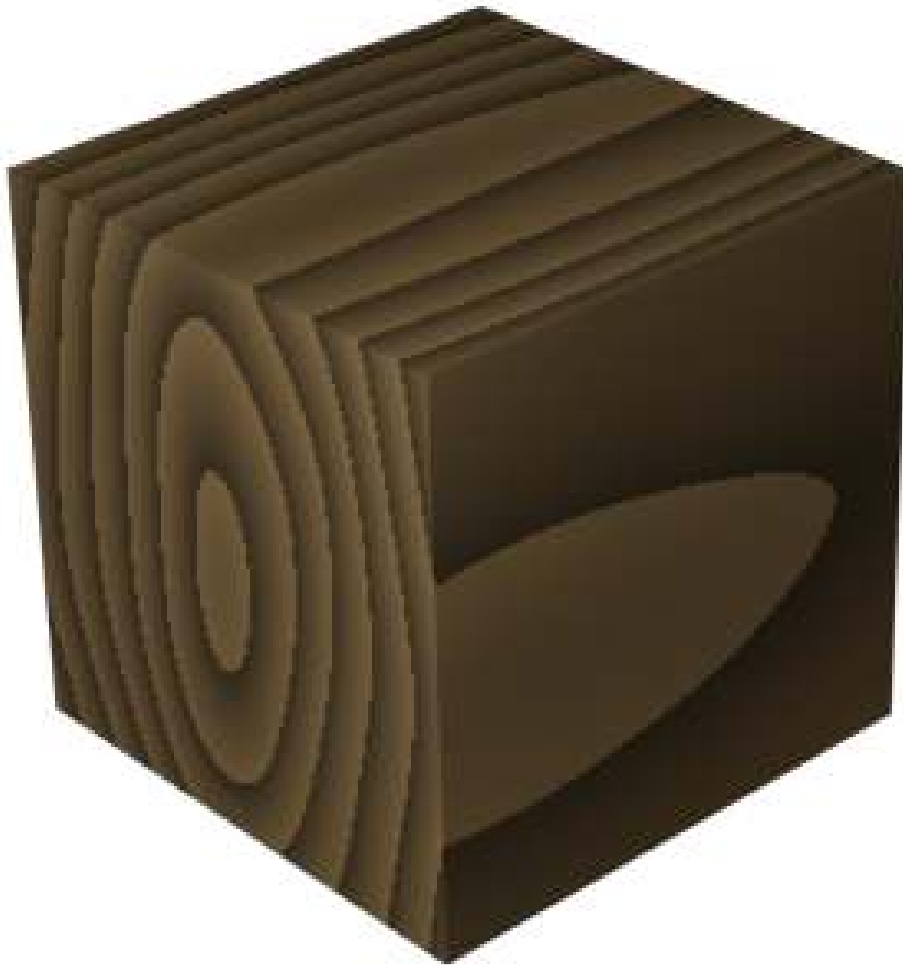


Figure 160

- Remember that you need to Render to see the wood the grain on your object.

40. Quickie Texture

Information

Applicable Blender version: 2.49

Textures are laid on top of materials to give them complicated colors and other effects. An object is covered with a material, which might contain several textures: An image texture of stone, a texture to make the stone look bumpy, and a texture to make the stone deform in different ways.

A texture may be an image¹ or a computed function². What the texture does and how it is mapped onto your object is set in the material buttons. Some commonly used texture types are shown on the page Using Textures³.

This tutorial uses the file from the **Quickie Material**⁴ tutorial. If you didn't do it before, go back and do it now.

40.1. Adding a texture

The file we're working with has a texture already. However, if you **add a new material** to an object, it has no texture by default.

To **add a texture**:

- In the *Texture* panel of the *Material* buttons click on *Add New* to add a new texture, or select an existing one with the Drop-Down button.
- Select the *Texture Buttons* with the spotty square icon

1 Chapter 38 on page 227

2 Chapter 39 on page 231

3 Chapter 72 on page 655

4 Chapter 36 on page 213

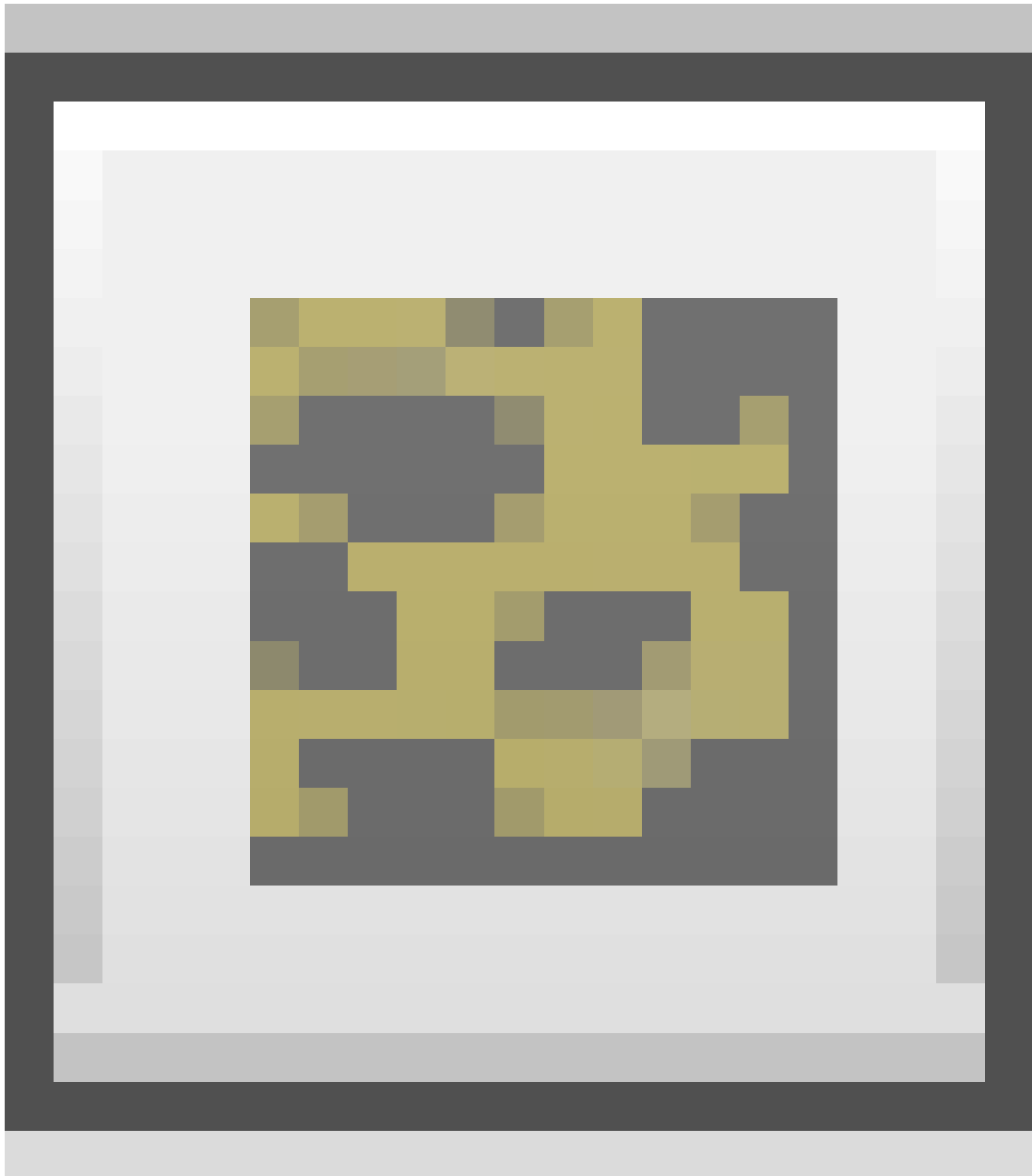


Figure 161

(or by pressing **F6**).

- Select the texture type *Clouds* from the drop-down list. You can also change the texture's name, as we have done for the material.

Blender 2.58

- Click on *Texture* in the buttons panel.
- Click on the + *New*-button under the list of textures.
- Select the texture type *Clouds* from the drop-down list. You can also change the texture's name, as we have done for the material.

A preview appears, as well as some parameters to experiment with. A *Clouds* texture provides some irregularities.



Figure 162 Image 1a: A simple Clouds texture

--- Note: In blender 2.58 all of the following settings are found under "Texture->Influence"---

Using Blender 2.61 will have difficulties in completing this task due to major UI differences

- Head back to the *Material Buttons* (Click the red sphere or press **F5**) and a colored preview of the texture appears. It is purple! All new textures default to this colour.

On the right hand side of the material buttons window there are three tabs: *Texture*, *Map Input*, and *Map To*. Since we want to change a material property that is affected by the texture we have to look at the *Map To* tab. This means: map the value that a texture provides to a material property. A texture may provide different values:

1. RGB color (all images, *Magic*, every texture with a colorband)
2. Intensity, either as grey scale or/and an alpha value (most of the procedural textures, image textures with alpha, textures with a colorband)
3. Normal values (*Stucci*, normal maps)

If you want to use textures you always have to be aware of the value a texture provides and the *Map To* settings for the material.

- Select the *Map To* panel.

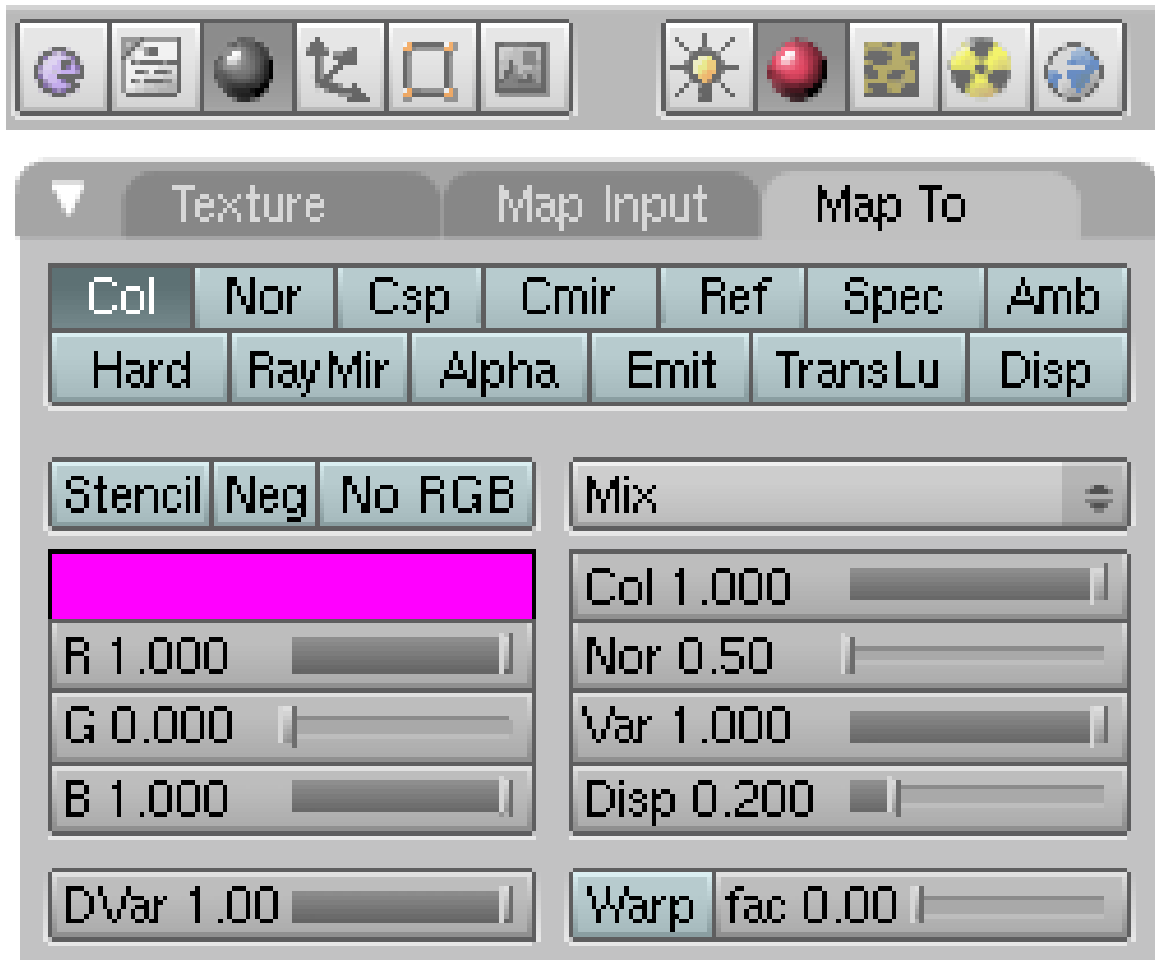


Figure 163 Image 1b: Default settings in the *Map To* panel

The activated *Col* button in the top row shows that the texture affects the color of the material. A *Clouds* texture provides an intensity value, ranging from 0 (where the texture is black) to 1 (where the texture is white).

The RGB (Red, Green, Blue) sliders here adjust the target color of the texture. This is mixed (*Mix*) with the material color, where the intensity of the *Clouds* texture is 1 only the target color is used, where the intensity of the *Clouds* texture is 0, only the material color is used.

- Set the target color to black.

Procedural textures are not shown in the 3D window (only if you would use excessive amounts of vertices), even if you use GLSL materials. This means that we have to render the image to see the texture properly.

- Render the image (**F12**), or use the preview to judge the result of the texture.



Figure 164

Next we will add a *Stucci* texture to make our clouds look bumpy.

40.2. Adding a Stucci texture

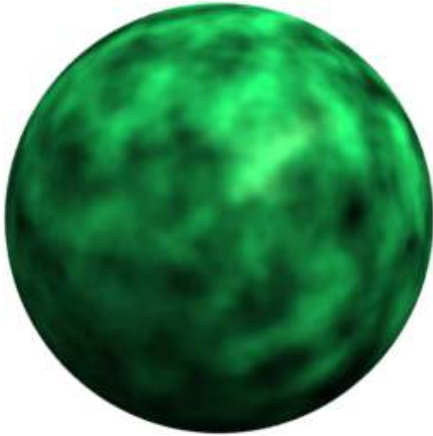


Figure 165

- Go back to the *Texture Buttons* and select the next texture channel (one of the blank buttons under "NewCloudTexture").
- Add a new texture here and set the type to *Stucci*.
- Back to the *Material Buttons*, and click the "Map To" tab.
- Turn *Col* off and *Nor* on.

Col means the texture affects the colour. *Nor* means it affects the rendered normal, i.e. the angle the renderer treats the surface as - creating fake shadows on the surface.

- Play with the *Nor* slider, but leave it on about 4.

In blender 2.58

- Go to the *Texture* button and select the empty texture beneath the one you just added.
- Add a new texture and set its type to *Stucci*.
- Go to Influence and turn off *Color* and turn on *Normal*.

In blender 2.6x

- Go to Influence and deselect *Diffuse*. Select all *Geometry*.

Render to see the effect. A texture changing the surface normal is called a "Bump" map or "Normal" map. Since that is fake 3D, you don't see it under every circumstance, you will get a greater effect on smoothly curved surfaces with high specularity, only a little or no effect on flat surfaces with low specularity.

41. Halo Materials

i Information

Applicable Blender version: 2.48

41.1. Introduction

Halos are a neat effect that you can use in Blender. Instead of affecting the color/texture of a *face*, like normal materials do, when you use a halo material, it affects the *Vertices*. The particles with *Point* Visualization are rendered as Halos also.

Here is an example.

This is the standard Blender start file rendered as per normal, with a regular material:

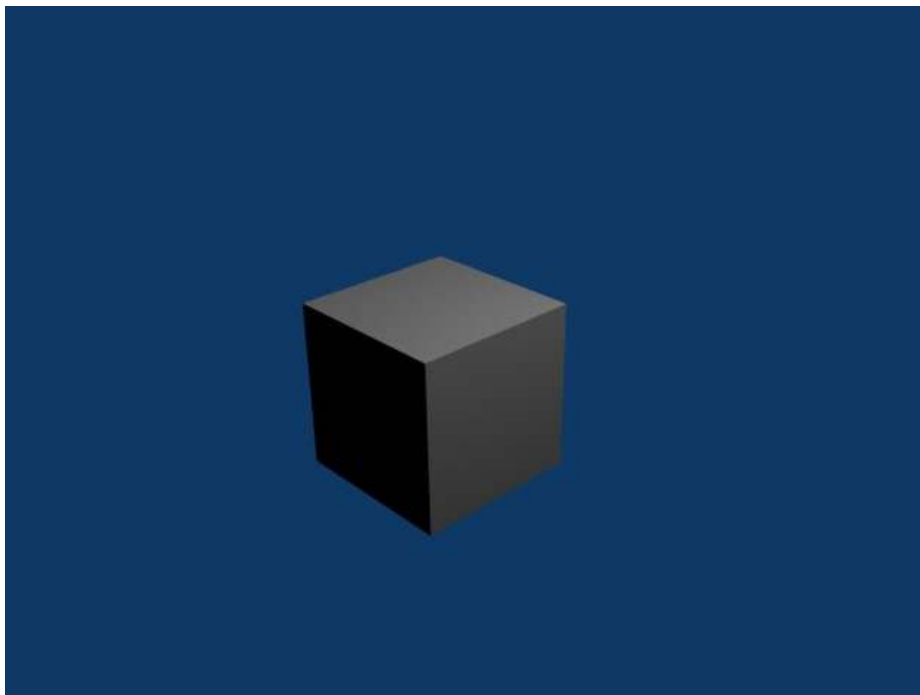


Figure 166

This is the same scene, rendered with a basic Halo Material:

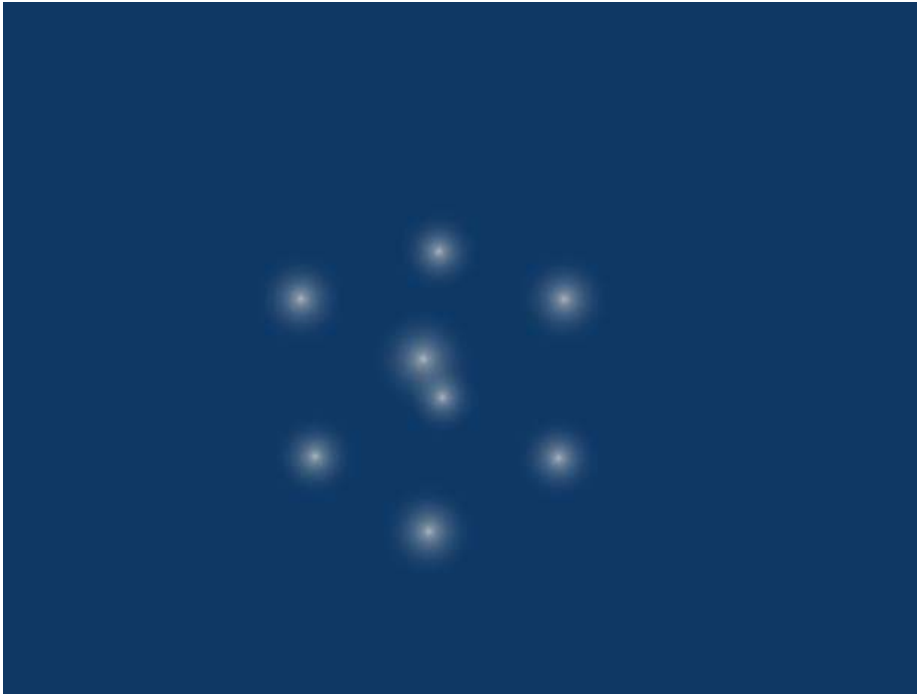


Figure 167

As you can see, it gives the vertices a 'star'-like appearance, and can be used to make some neat effects.

41.2. Applying a Halo Material

Blender before 2.58

By now you should know how to basically use the materials panel. If not, please complete the **Quickie Material¹** tutorial.

To apply a halo effect, simply click the Halo button in the Links and Pipeline panel. The buttons highlighted below are what you need to click to get to this panel (plus the Halo button)



Figure 168

¹ Chapter 36 on page 213

The Preview panel will now show the halo effect on the sphere:

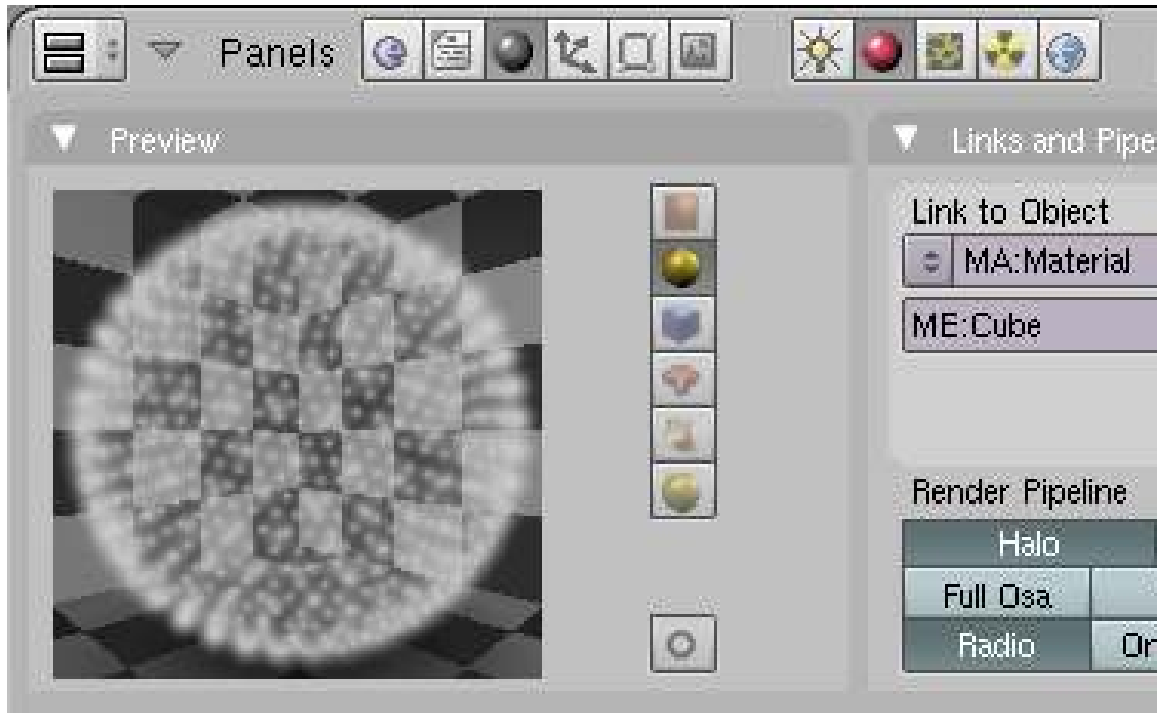


Figure 169

It's as simple as that!

Blender 2.58 - 2.63

- To apply a halo effect, simply click on the Halo button in the Properties panel (see below):

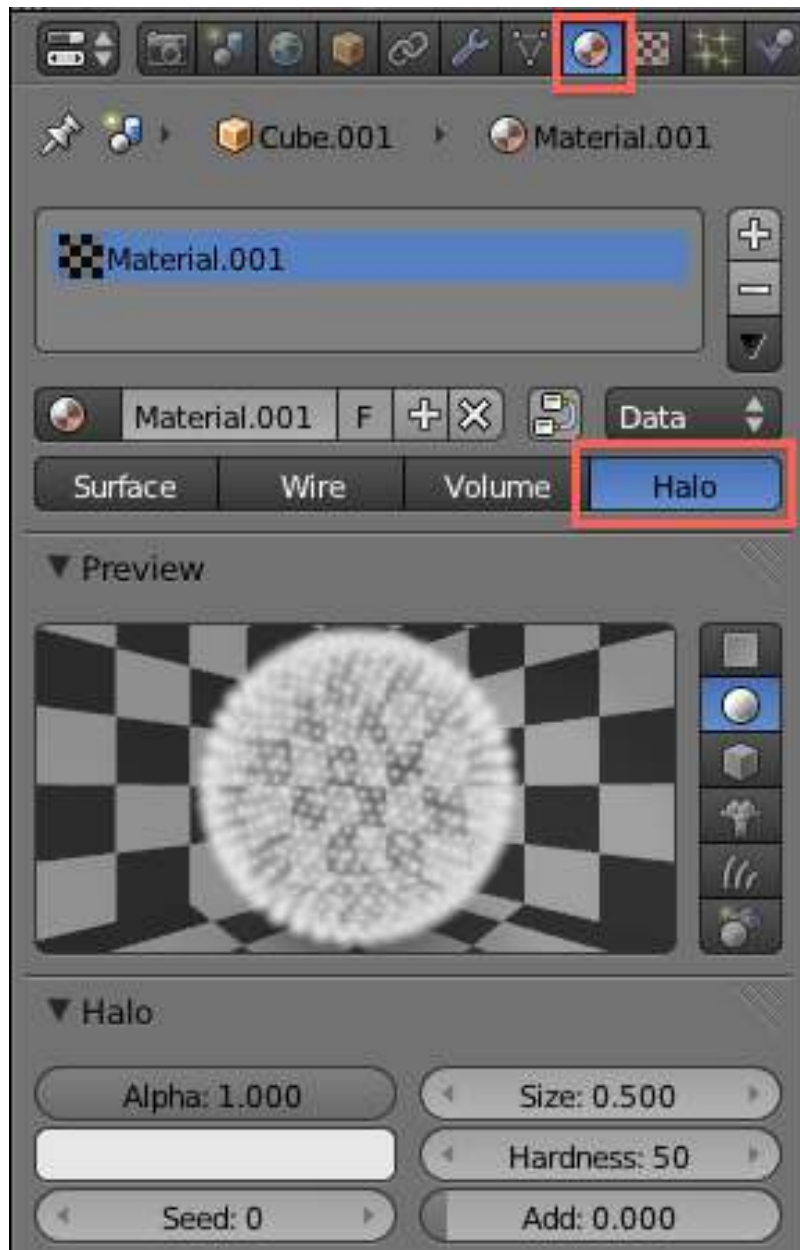


Figure 170 Halo properties (Blender 2.61)

(Halo properties - Blender 2.61 screen)

41.3. Adding/Modifying extra effects

Now you know how to apply halos, let's look at some of the settings for them.

Note that when you clicked the Halo button, the color panel to the right changed slightly.

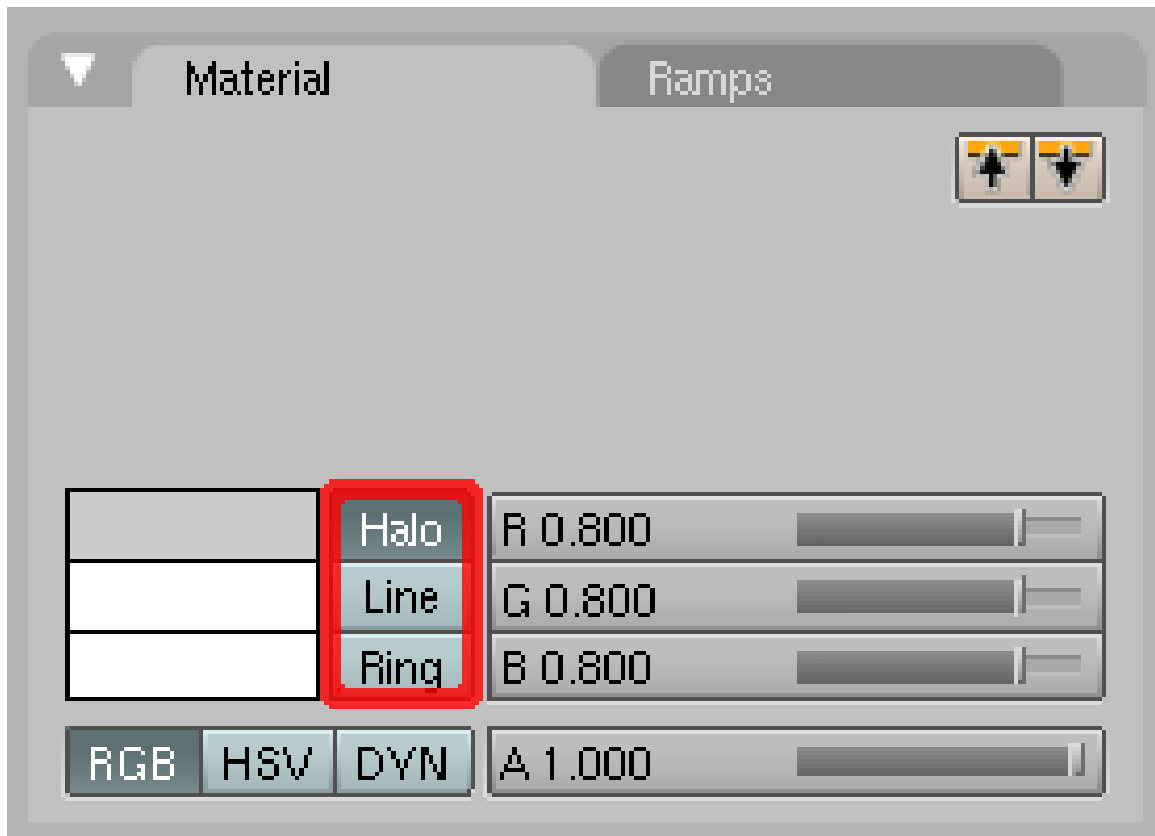


Figure 171

These settings will now allow you to change the colors of various aspects of the halos. Have a mess around with the Halo color and you'll see that it affects what you currently see in the preview panel. The other 2 buttons, **Line** and **Ring**, will affect other elements of the Halo we'll look at now.

Blender 2.58 / 2.61

You got a new panel called *Halo* when you've clicked the Halo button. All settings mentioned above can be found there.

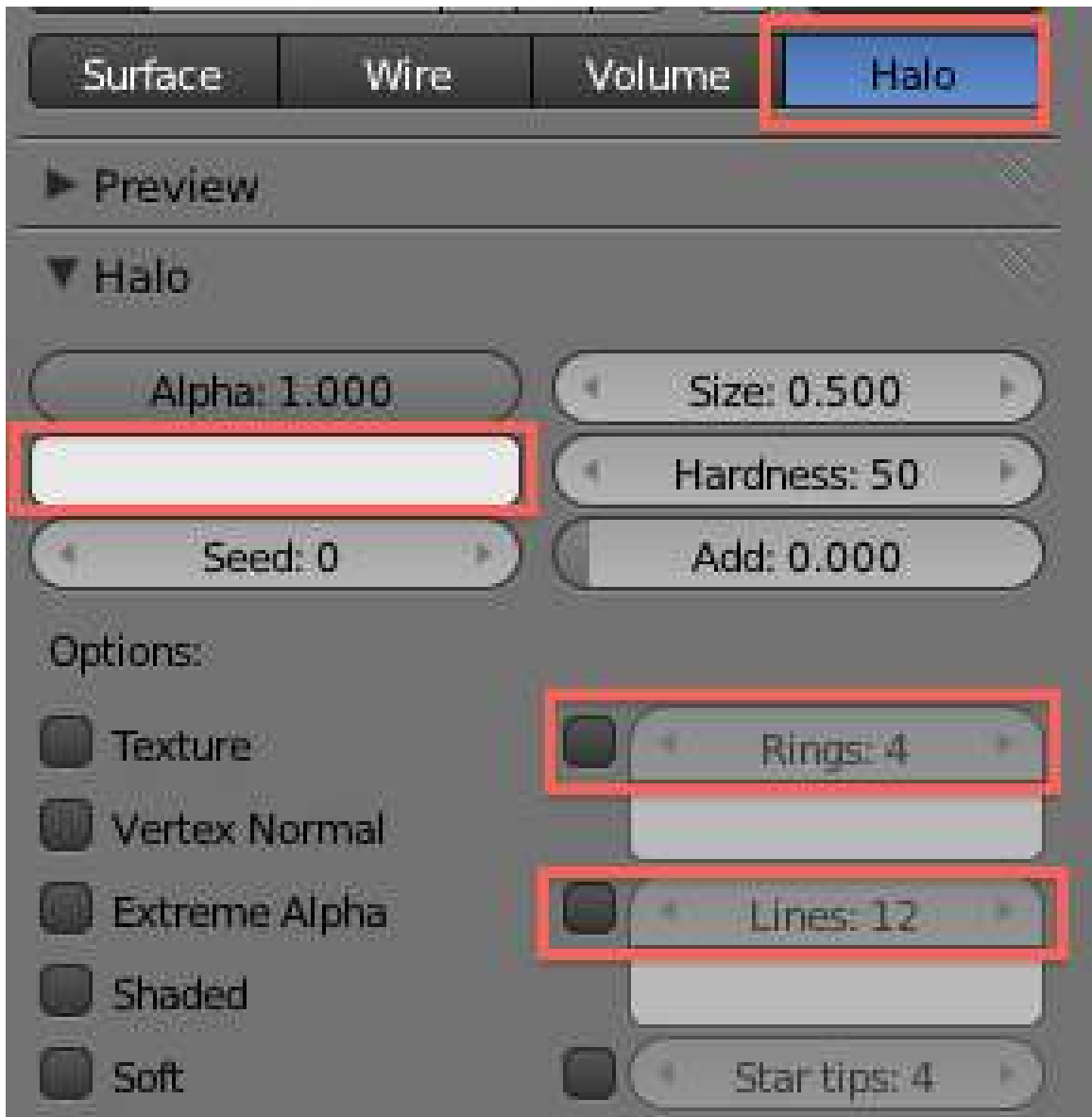


Figure 172 Blender261-Halo-Properties-Screen-2

(Halo properties - Blender 2.61 screen)

Look over to the right of the bottom panels, and locate the 'Shaders' tab. By default, it should be next to the 'Mirror Transp' tab.

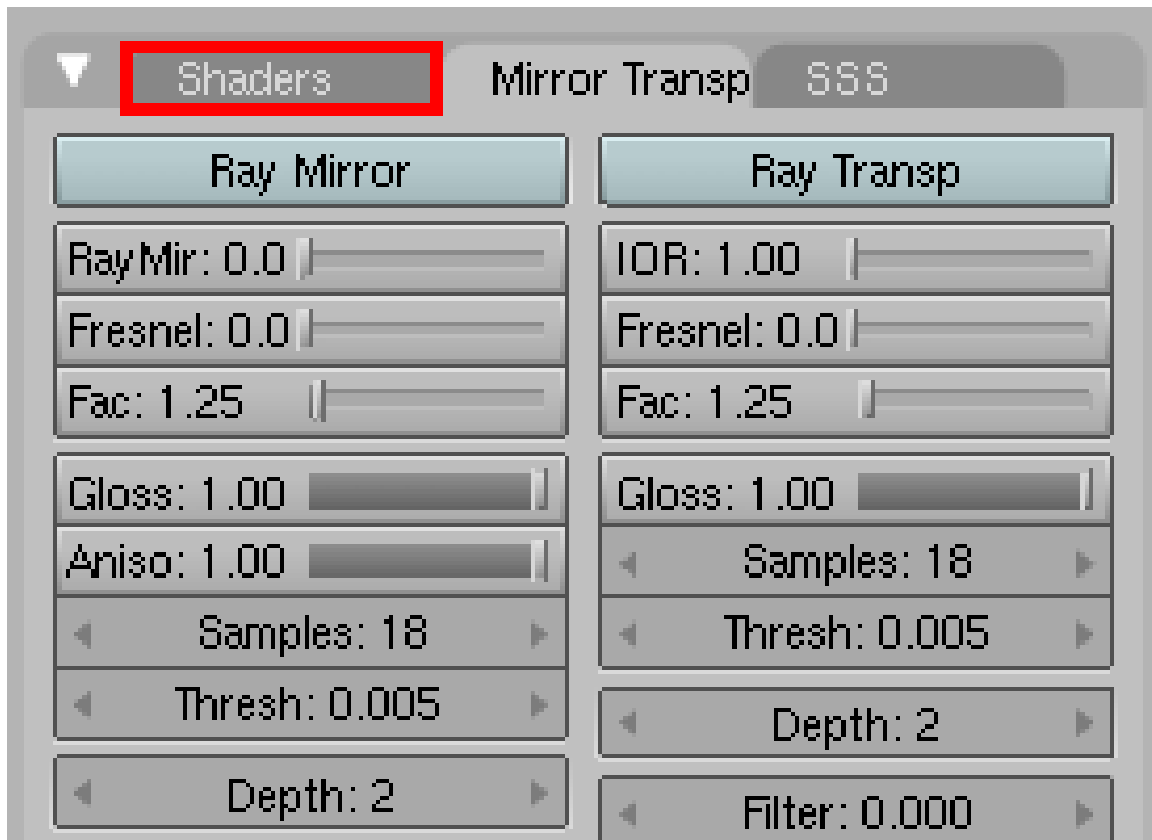


Figure 173

You can now see all the various options for the halos:



Figure 174

You can easily modify the size of the halos with the HaloSize box and the 'Hard' slider modifies the 'fuzziness' of the halos. The 'Add' slider controls how much the halo is 'added' to whatever is behind it - this is more used for textures to control how much of the texture you want to appear on the objects. An easy way to think of the 'Add' slider is as a transparency for all of the halos.

Here is a run-down on the buttons to the right:

Flare (2.58 - Panel called *Flare* beneath the Halo-panel)

Adds lens flare - a technique which can add a sense of realism to an animation. By default it seems to just make the halos seem brighter, but if you modify some of the settings that appear down the bottom you can get some pretty wacky effects, especially if you have the vertices moving around in an animation.

Rings

Adds rings around the halos. Number of rings can be modified in the 'Rings' number panel.

Lines

Adds lines coming off the center. Number of lines can be modified in the 'Lines' number panel.

Star (2.58 - Star tips)

Gives the halos a star shape. The amount of points on the star can be changed to the right in the 'Star' number panel.

HaloTex (2.58 - Texture)

Allows the Halos to be rendered with a texture. This can create some very cool effects.

HaloPuno (2.58 - Vertex Normal)

Lets the vertices' normals specify the size of the halos. Only becomes really obvious on complicated meshes. For a good example, have a look at what it does to Suzanne the monkey.

X Alpha

intensifies the transparency effect of the halos.

Shaded

Allows the Halos to be shaded and their color affected by light sources.

Soft

Softens the Halos

You can now see where the other color selectors from before work. They let you modify the color of the lines and circles as well.



Figure 175

Blender 2.58 / 2.61

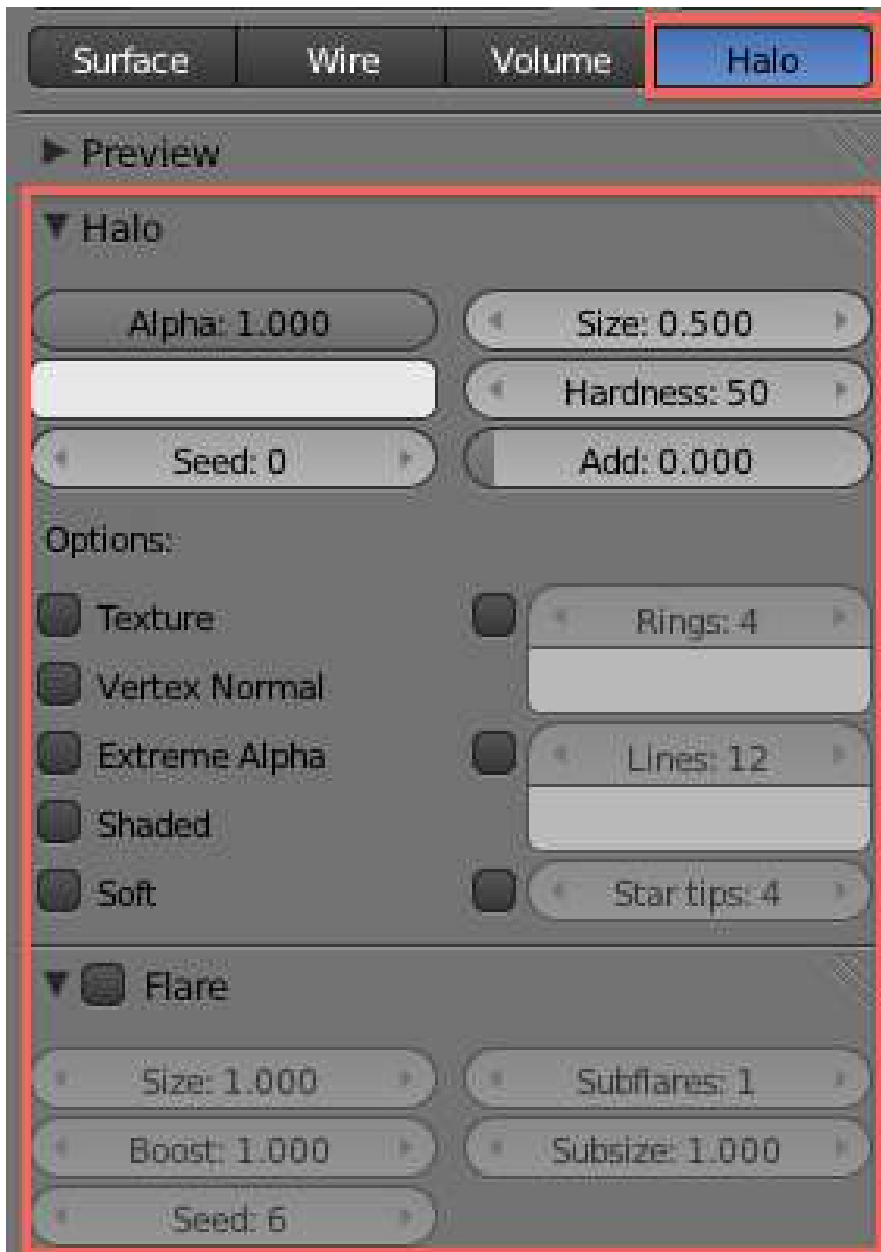


Figure 176 Blender 2.61 Halo Properties Screen 3

(Halo properties - Blender 2.61 screen)

42. Aligned to view issue

After much struggling to follow many tutorials based on older versions of Blender, I have downloaded multiple versions to discover why the tutorials based on versions such as 2.43 don't work when attempted on updated versions such as Blender 2.48a and above. Newer versions such as 2.48 have added a new option to **/not/** have added objects rotated to the current viewpoint. With older versions, being in top, front or side view would cause any newly added objects to face different directions on creation.

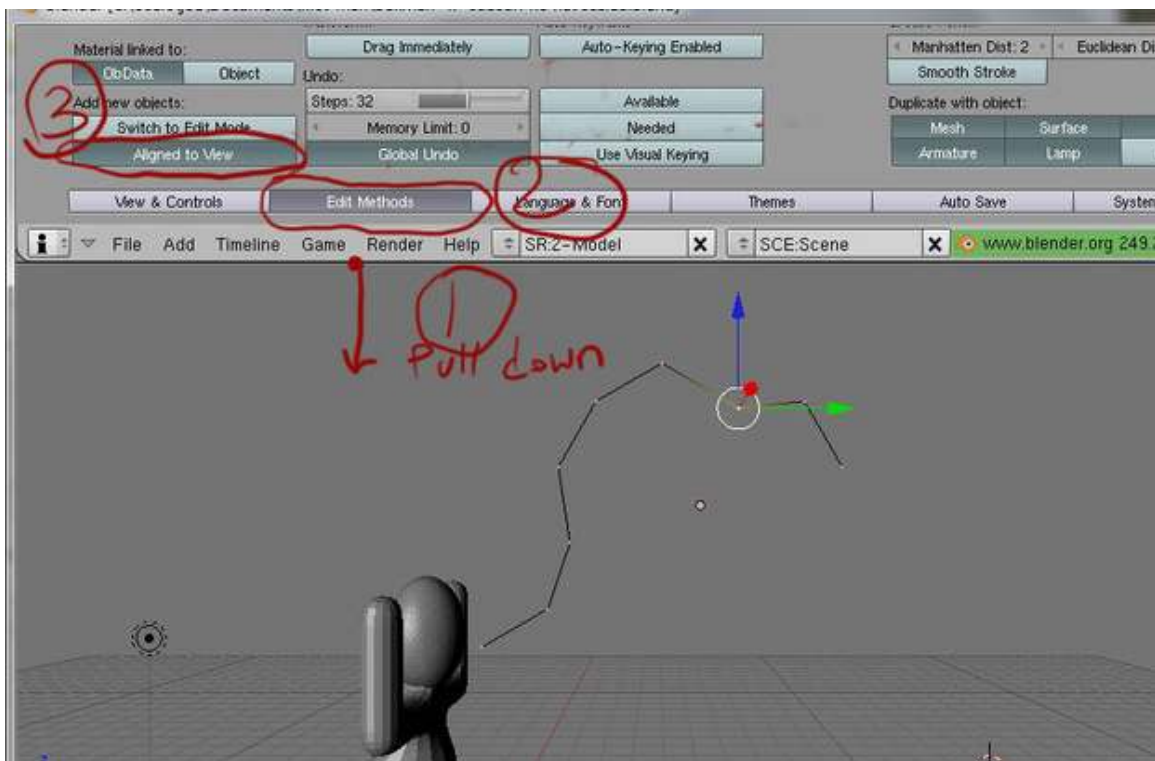


Figure 177 The location of the button to make the circle show up in the correct orientation

Newer versions of Blender introduced the ability to force all objects into the same global orientation; even worse, they set it up that way **BY DEFAULT!** This means that unless the user deliberately changes the settings in the new versions, many older tutorials will act as if they are broken.

Newer versions of Blender (such as version 2.48, or 2.49b) can be set to act in the same way as the older versions, by setting the **Align to view** on the (i): **USER PREFERENCES** menu in the right way.

Making these simple changes will "unbreak" tutorials written under Blender version 2.43, by allowing new objects to be automatically oriented to whatever viewscreen orientation is selected in the active viewscreen.

Any time object rotations, lattices or whatever else end up completely out of alignment with what older tutorials say should happen, these steps are your first best fix for almost every such situation.

In 2.58 and 2.61

The settings mentioned above are found in "File->User Preferences" (shortcut: Ctrl+Alt+U) under the "Editing" tab. There's a drop-down called "Align To" where you can set "View" or "World".

Noob Note: What actually happens by default on newer versions of Blender is that the axis of rotation is perpendicular to the screen. It means that, instead of revolving around the vertical axis, the object will revolve in the plane of the screen. Another way to deal with this is to change the view just before performing the rotation (I used **NUM1** view) and come back to **NUM7** once done.

43. Bones

Bones are used for shifting models and making them posable. If you are not ready for this yet and wish to continue simply modeling, please skip this tutorial to the next section.

Bones are a modeling tool that are especially important for animating characters. Bones allow you to move characters' limbs in a way that is much simpler than trying to re-arrange the vertices every time.

It works by associating a bone with particular vertices, causing them to move along with the bone when the position is changed in pose mode. Using bones is fairly simple once you get the hang of it, but, like many things in Blender, can be a little daunting at first sight.

Bones don't do much on their own; in fact, they turn invisible at render time. For this following module we'll use the previously made character from module [Modeling a Simple Person](#)¹. Note that while we will be using bones on a simple person, the process can be used with any creature or body type you imagine!

¹ Chapter 30 on page 167

43.1. Laying down bones

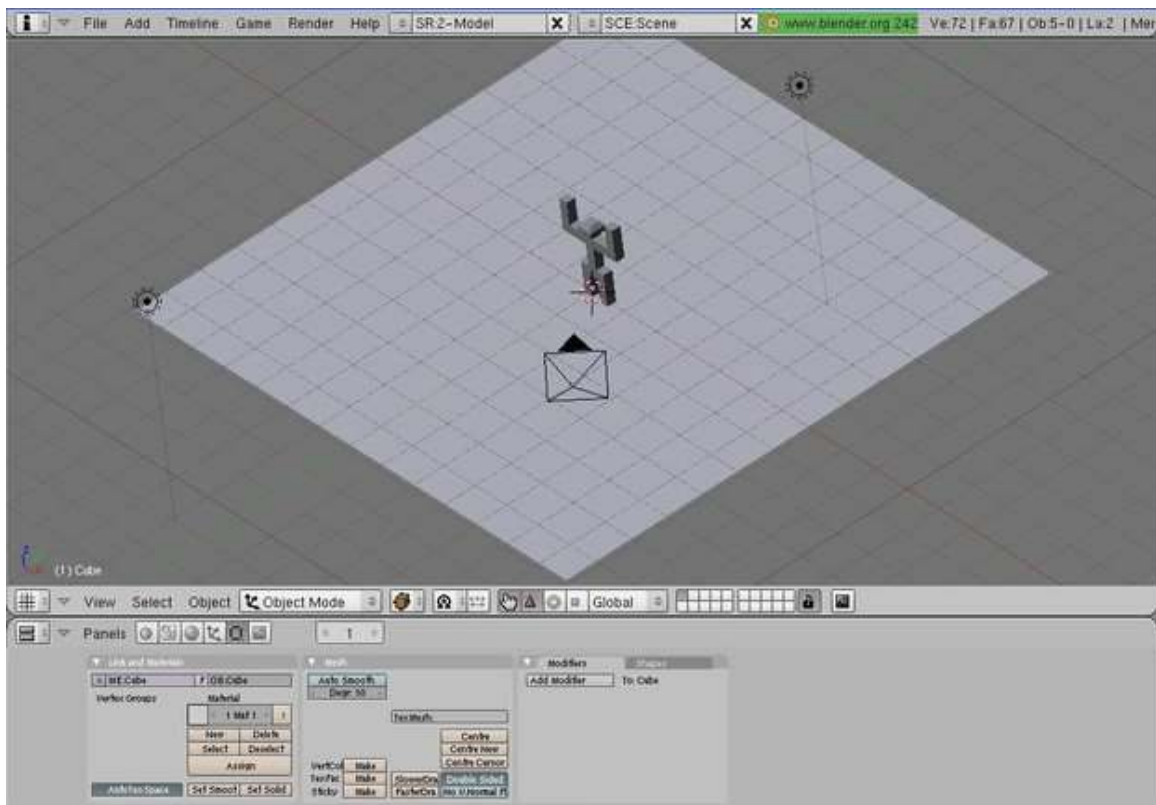


Figure 178

Note: This just shows the basics of adding bones to an object. Go to the advanced animation² page for a more comprehensive guide on this.

First of all, we'll need a model to put some bones on! For this tutorial, we're going to use a humanoid model. I'm using a quick model that I made based on the Blender 3D: Noob to Pro/Modeling a Simple Person³ first tutorial. It's rather blocky, but this isn't a detailed tutorial. However, you can find a model pre-made just for you [here](#)⁴.

Okay, first of all, here's our setup, with Block Dude standing on the plane.

Noob note: You will be placing armatures ("bones") inside your humanoid, so you must work in "wireframe mode," not "solid mode." Otherwise, you will not be able to see the armatures when you place them. To toggle between "solid" and "wireframe," press **Z**.

Note: An alternative to working in "wireframe mode" is to turn "X-Ray" on for the armature. To do this, select the armature and go to the editing menu (**F9**). Under the armature tab, click "X-Ray" in the top row of buttons. This will allow the armature to show through other objects.

² Chapter 152 on page 1157

³ Chapter 30 on page 167

⁴ <http://www.nusoy.com/blender>

43.2. Add a bone



Figure 179

Now, let's put some bones on Block Dude! In Object Mode go to **[SPACE]** → *Add* → *Armature*. (Blender 2.5x note: **[SHIFT]** + **[A]** → *Armature* → *Single Bone*)

What we are looking at is an armature. This is a single bone. Now, we need to put the bone in Block Dude! Move and rotate the bone so that it's in the middle of Block Dude's chest.

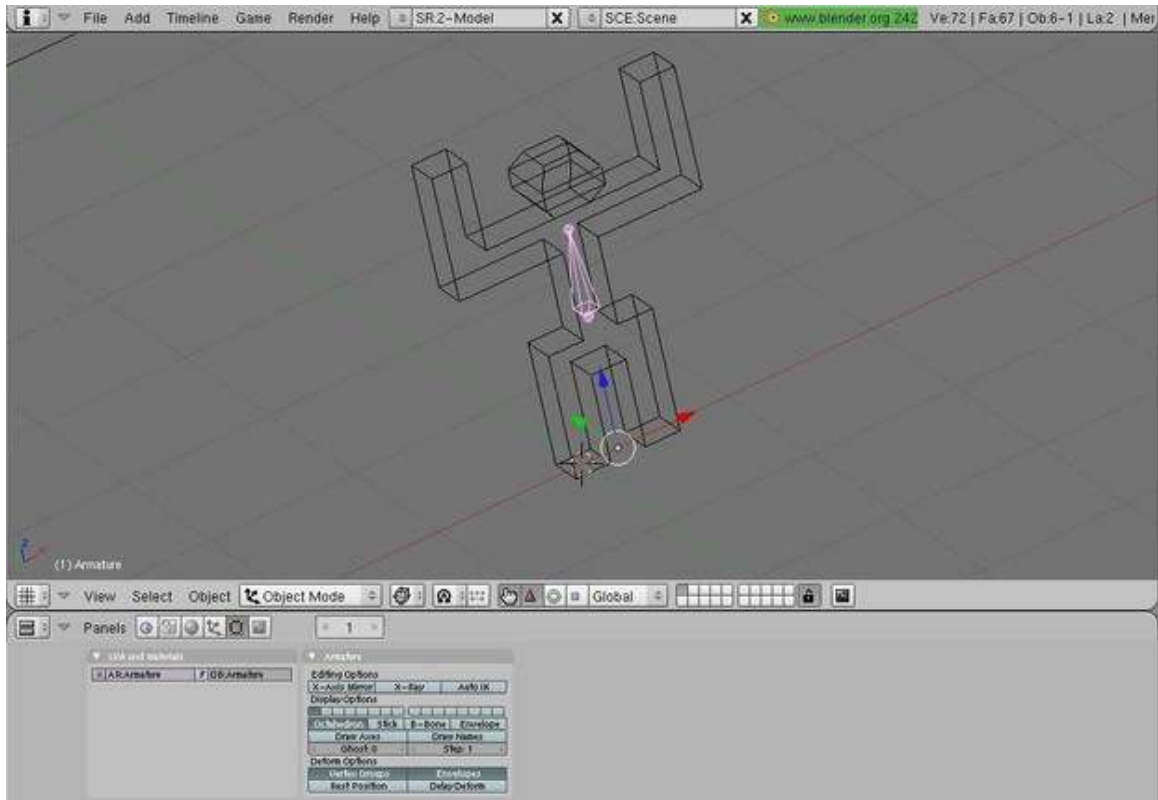


Figure 180

If your bone does not have the correct length, then change the size of the bone by moving one of the ends of the bone : switch to Edit Mode, select one of the ends of the bone, then move it using **G** .

43.3. Extrude a second Bone

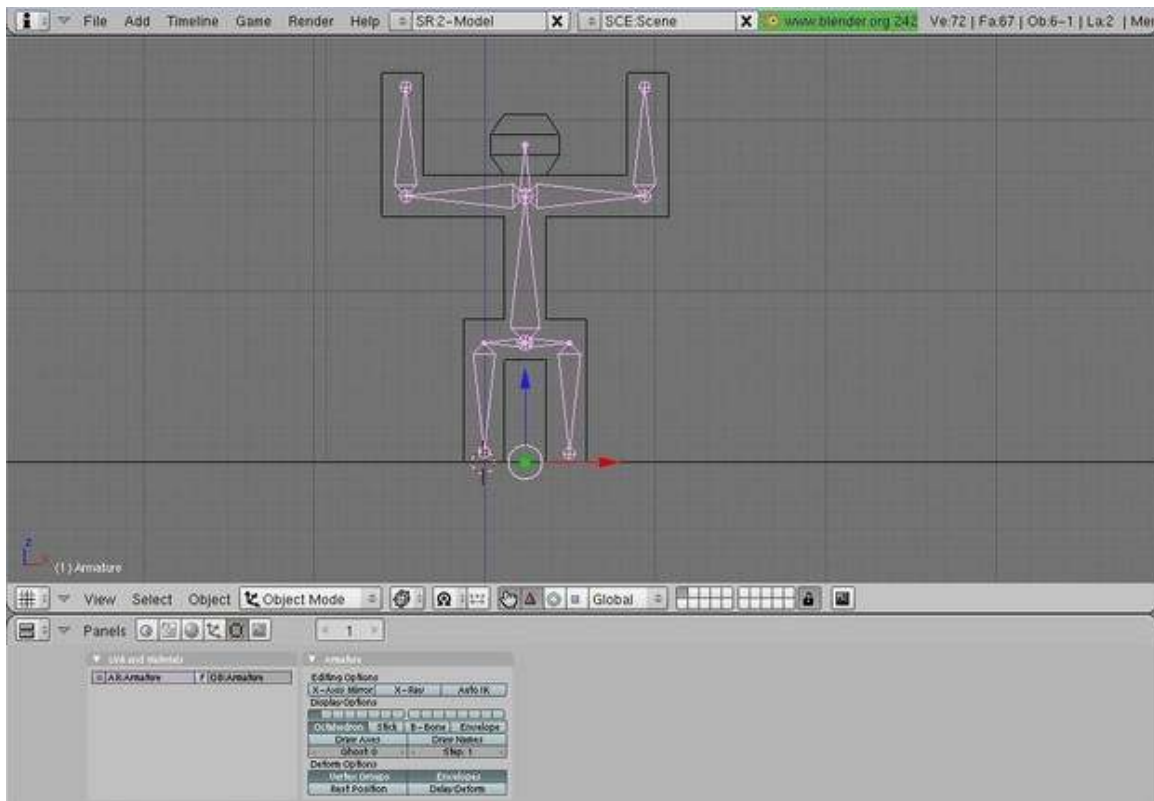


Figure 181

To create a second bone starting from one of the ends of the first bone, make sure you are in Edit Mode, select the end of the first bone, then press **[E]**. The second bone appears, with its start point on the selected end of the first bone. Move the mouse to position the end point, then press **[LMB]**, **[ENTER]**, or **[SPACE]**. Now, extrude (**[E]**) and scale the bone as needed to put the bones in his body! Make sure that you are in Edit mode, and click the pink dots at the end of the bone to do things like extrusion. These operate much the same way as vertices, you can extrude, rotate, move, and even subdivide. Your finished result should look something like this:

43.4. Name the bones

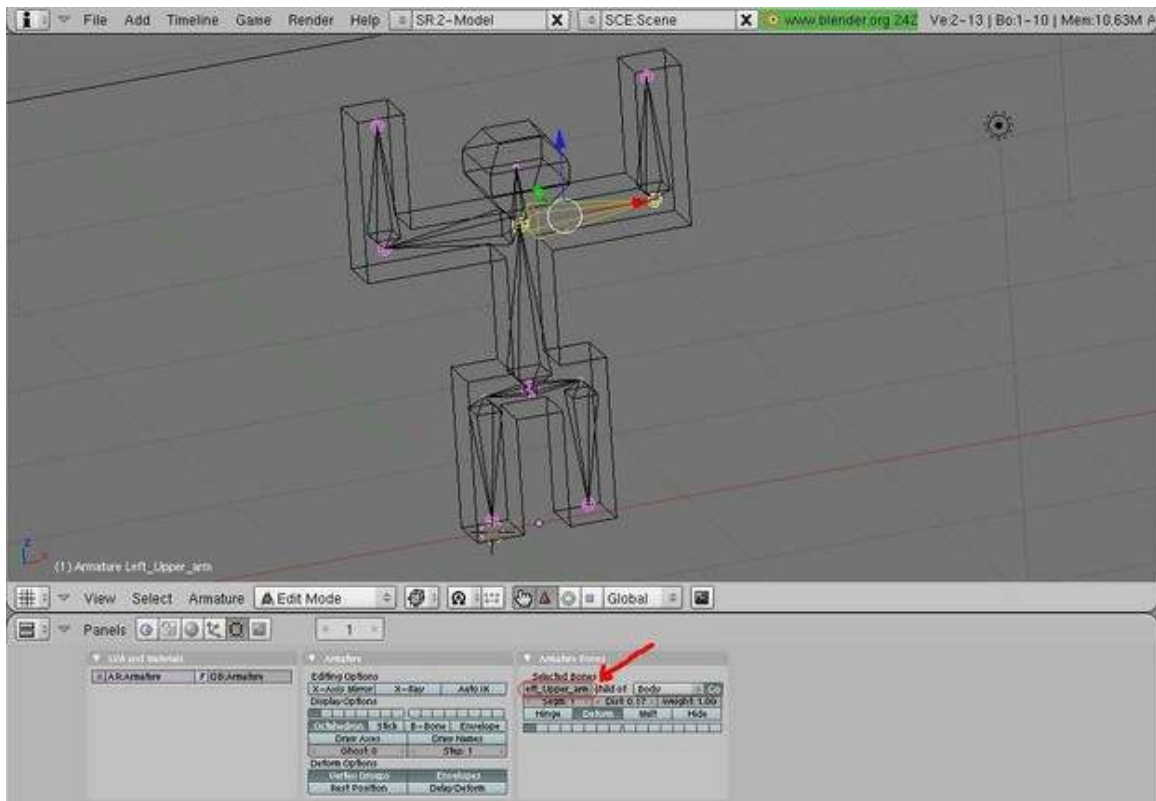


Figure 182

Now, just to make things easier, we're going to name the bones. For example, my bones are named RT_Forearm, Left_Forearm, RT_Upper_Arm, etc. Select the bone and press **[F9]** to display the Editing panel. In the "Armature Bones" sub-panel click the top left box to edit the name, indicated with a red arrow in the screenshot below. (Noob note: when you are naming the bones remember that if you are looking at the person from the front, your left is the person's right. To make the naming easier switch to viewing the person from behind using **[CTRL] + [NUM1]** .)

Blender 2.58

This is most easily done in the "outliner" (top right by default) by selecting each bone and renaming it under the "Bone" button in the Buttons-Panel (the button looks like a dog bone). You can select the bone and see it highlighted in the 3D bone, then **[CTRL] + [LMB]** to rename.

43.5. Parent the bones

Now, we need to parent the bones to the mesh. Go back into Object Mode and select Block Dude. Now, select the Armature as well, and press **[CTRL] + [P]** , a menu will pop up, select **Armature**

(**Armature Deform** in 2.6x), then **Create From Bone Heat** (2.46+) (**Create From Closest Bones** before blender 2.46)(**With Automatic Weights** 2.5x and 2.6x).

Noob Note: the selection order is important in defining which object is the parent, so you cannot select both objects at the same time. You must select the armature last to make it the parent.

43.5.1. In depth info on selected bone topics

Add/remove mesh from bone control

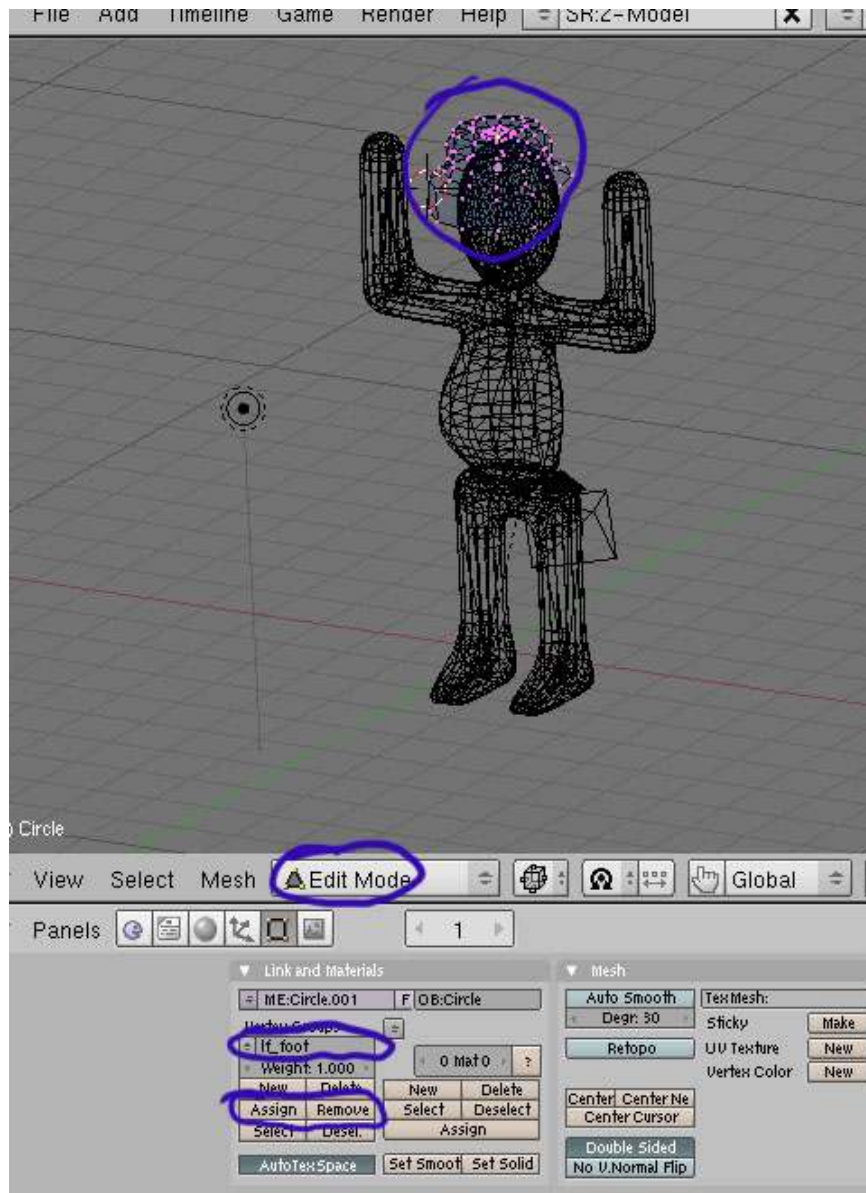



Figure 183

Create From Bone Heat creates mesh areas within the parented object that are associated with armature bones. This is done automatically, according to the bones' position and size. Alternatively, choose **Name Groups** instead of **Create From Bone Heat** if you don't want Blender to assign groups automatically, and do it manually.




To manually change the mesh areas that the bones control, go to **object mode**,  on the **MESH** you want to add/remove (if it's inside the same object, then select the areas of the mesh you want to work with), go to Edit Mode, and make sure you're looking at the edit mode buttons in the bottom button window.

You should now see on the left a link and materials panel with a pair of Assign and Remove buttons(you may need to scroll with the middle wheel while hovering over the bottom button windows to make it appear, if it's off to the left of what you can see). In Blender 2.59 activate in the properties window the 'Object Data' and you will find the relevant data in the subwindow 'Vertex Groups'.

Now pick the bone group from the dropdown above the assign/remove buttons, and then hit assign, or remove. Usually vertices will be in one group, but can be assigned to multiple groups.

Mesh deforms like it's far away from the bones

if the mesh is properly assigned to the bones they will move regardless of whether the bones are inside the cylinder or not (HOW they deform WILL be affected however). The most common mistake in this step is creating and (more importantly) parenting the cylinder to the armature while it is outside the cylinder, which causes Blender not to assign vertices to any bones at all. You can check this by editing the cylinder, selecting a vertex group in the **Links and Materials tab**, and pressing **select**. This will highlight the vertices associated with the bone. If none are selected, it means none were assigned in the first place - in which case you need to assign them manually as explained above.

If there is no effect, in Edit or Pose mode select that bone (or bones) and choose Envelope display mode ( → *Armature* → *Envelope*), then press  +  and increase its area of influence to cover all faces that should be influenced by the bone.

43.6. Moving the Bones

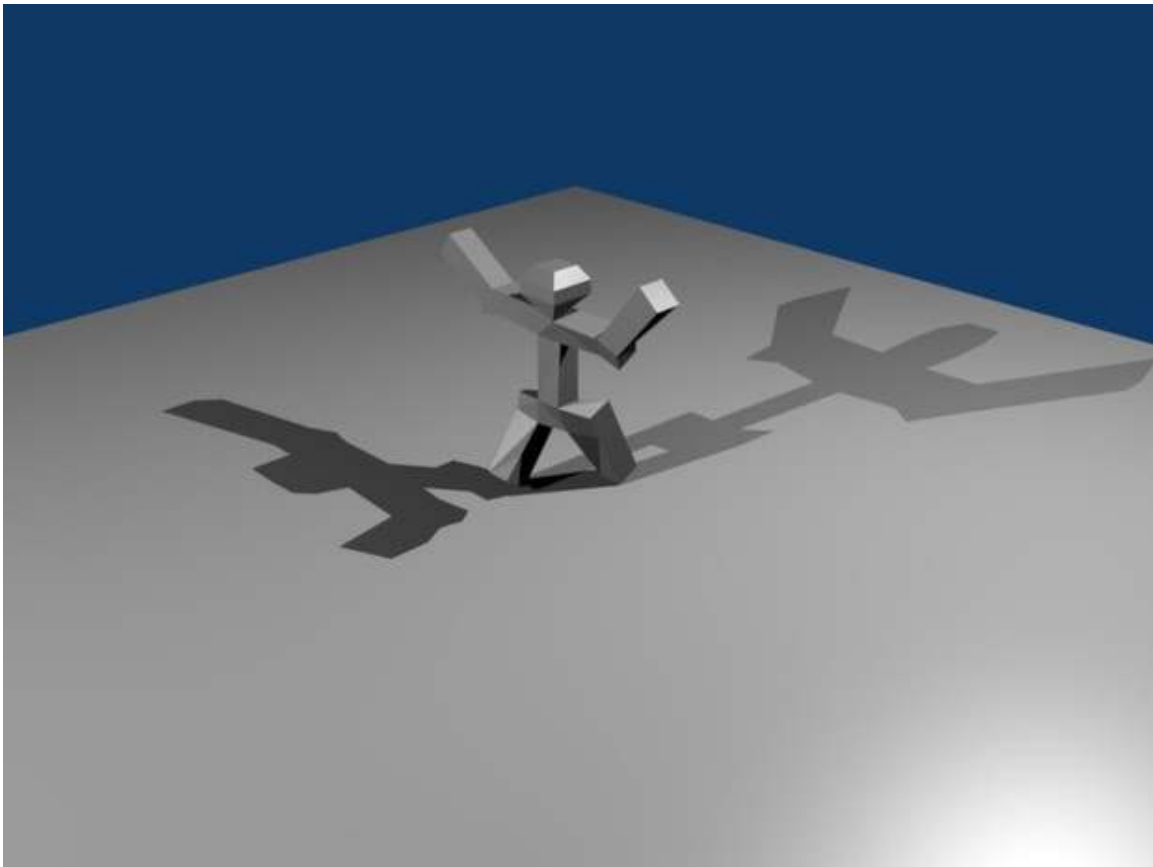


Figure 184


Go into **object mode**, RMB on the armature to select it, then enter **Pose Mode** (**[CTRL] + [TAB]**) (you need to select the armature for this option to appear!). Try moving a bone around by **[RMB]** to select it, and then hitting **[G]** , or **[R]** to move it.

If you've done everything correctly, your mesh should move when you move the bones! If this doesn't happen, scale the bones up so that they fit better in the mesh, and scale up the bones until they do what you want (read comment in the parenting section above on adjusting the bones envelopes if you do not get an effect while moving/rotating the bones). With the bones now, you can put Block Dude into a lot of different positions without moving individual vertices.

To the right is an example of how you can move Block Dude with the bones.

To move individual bones, you have to go into pose mode. You will only see pose mode listed on the object/edit mode drop down when you have **[RMB]** on the bones from object mode. Also while in pose mode if after a **[RMB]** click you can't move bones with a **[G]** , or **[R]** , check the "Move Object Centers Only" button (just to the right of the Rotation/Scaling Pivot button).

Noob Note: If you've been adding bones to your simple person from the previous lessons, and have noticed the hat seems to stretch when you move the arms in pose mode, you will need to remove the hat from the arm mesh groups created in the prior bone heat step. To fix it, you need to go to

object mode,  on the hat, go to **edit mode**, locate the vertex group **assign/remove** buttons as described in the bones in depth step, Now pick the left arm, and right arm groups in the dropdown above the buttons, and then hit remove.

44. Mountains Out Of Molehills

Now that we've created our simple person, it's time to give him somewhere to go. In this tutorial we'll create a mountain range using a few simple, and handy tools.

44.1. Creating a simple plane

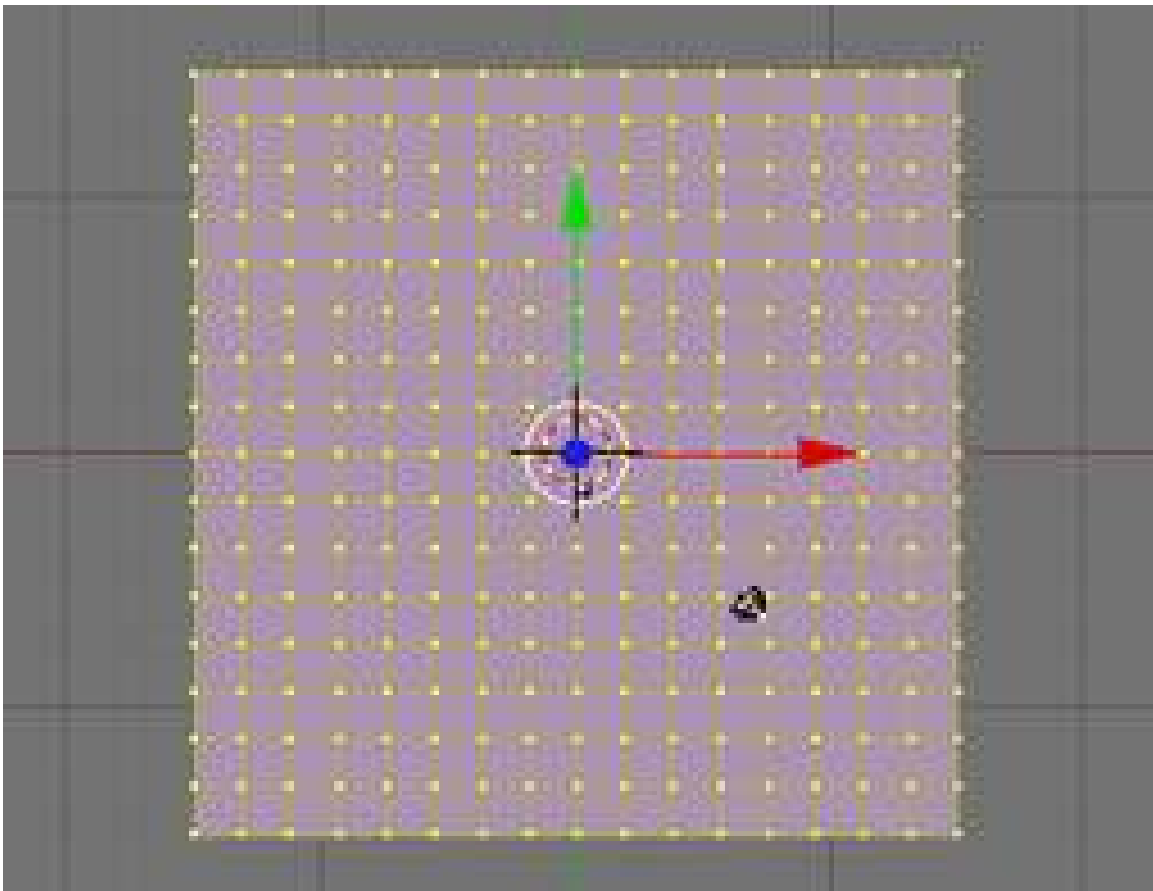


Figure 185

First we need a clean area to work with.

- Start off with a new project, using *File* → *New*, or hit **CTRL+NKEY**. If you have a default cube or plane just delete them now (select them with **RMB** and press **XKEY**).

Our first step is to create a large plane that we'll use for the ground and grow our mountains out of.

- Press on **NUM7** to enter top view. This way our plane will be lying flat when we create it.
- Press **SHIFT+C**. This sets the 3D cursor to (0,0,0) which will be the center of the plane we will add.(or use - **SHIFT+S** -> **cursor to center**)
- (Note: If you are zoomed in or out, pressing **SHIFT+C** will also reset the zoom view to its default setting)
- Now add the plane with **SPACE** → *Add* → *Mesh* → *Plane*. This will be our canvas. (**2.58** - **SHIFT+A** -> *Mesh* -> *Plane*)
- Change to **Edit mode** (**TAB** changes between edit and object mode)
- Scale the plane up by about 15. First put the mouse close to the center of the plane and press **SKEY** and drag the cursor away and watch the numbers in the bottom left of the 3D Window. Hold **CTRL** to increment by 0.1 for a more precise measurement. Alternatively, to enter the exact amount yourself press **SKEY**, then simply type **15** and hit **ENTER**.
- Now we need to add some vertices to work with. In the buttons window, make sure we have the *Editing* buttons open (or hit **F9** in the buttons windows to switch there).
- Under *Mesh Tools* hit the *Subdivide* button 4 times. Alternatively, in the 3D View window you can press **WKEY** and select **Subdivide**. (**2.58** - By default the *Mesh Tools* are on the left hand side.) You can also just hit **ENTER**, because *Subdivide* is the first option under **WKEY**.

Note: As in many of the next tutorials, you may choose to add a grid instead of adding a plane and then subdividing it to have vertices throughout the plane. Adding a grid has the added advantage of allowing you to select the number of vertices along the x and y axes (the two edges) of the grid at the instant of creating the grid, so there is no need to make further subdivisions.

44.2. First mountain

Now that we have the ground, it's time to start growing our mountains.

- Make sure you have nothing selected (**AKEY**).
- Select a random vertex with **RMB**. I usually start at the one that is 4 down from the top and 4 in from the left (the 4th vertex if you count the edges).
- Change to the side view with **NUM3**.

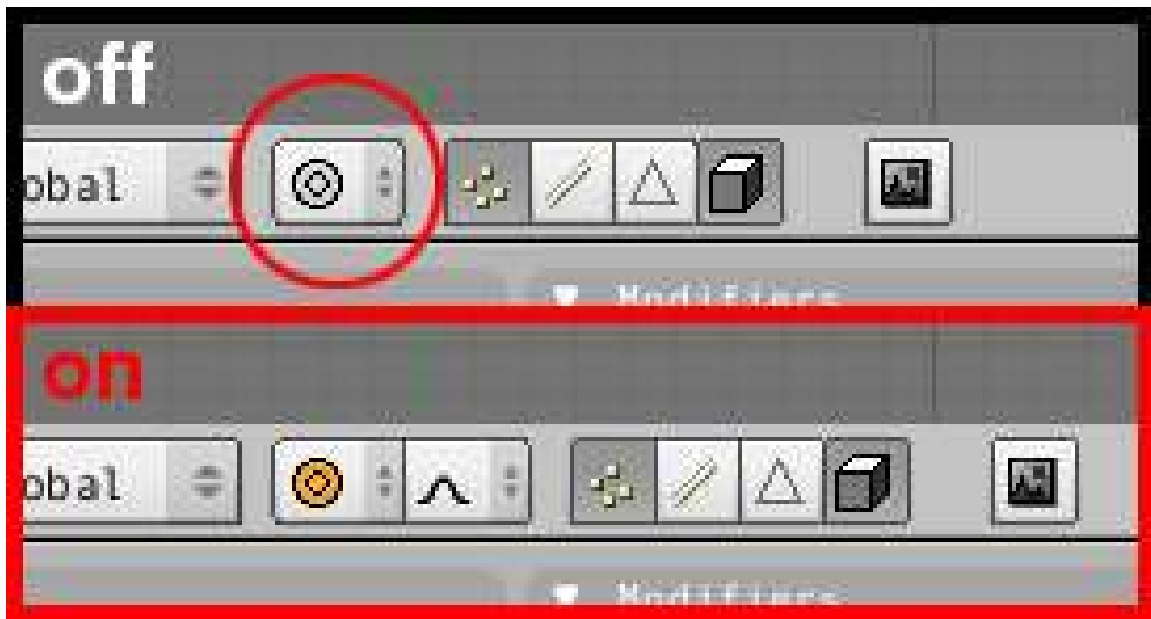


Figure 186

- Press **OKEY** to change to proportional edit mode or use the button which shows a grey ring on the header of the 3D View. The button will change its color to orange (blue in 2.61). You can also use **SPACE** → *Transform* → *Proportional Edit* (2.58 and 2.61 - By default this button is located just below the 3D view.)
- Once you've turned proportional edit mode on, another button appears to its right, the falloff button. Select Smooth Falloff here. Alternatively you can use the menu on the header of the 3D View (*Mesh* → *Proportional Falloff* → *Smooth*) or, using **SHIFT + OKEY** will switch between Sharp and Smooth Falloff (in versions prior to 2.37) or cycle through all 6 falloff types (in versions 2.37 and up) while using the Proportional editing tool.
- Press **GKEY** to grab the vertex. We should now have a circle surrounding the vertex, this is our *radius of influence*. Basically any vertices inside this circle will be affected by any changes to the vertex itself.

Noob Note: If you're having trouble seeing or changing the radius of influence, try saving your scene and restarting Blender.

- Use the **Mouse Wheel** or **PAGE_UPKEY** and **PAGE_DOWNKEY** to adjust the radius of influence to include just over 2 vertices on each side of our selected vertex. (Depending on your version of Blender, you may need to use **ALT + NUM+ / PAGE_UPKEY** and **ALT + NUM- / PAGE_DOWNKEY** and may need to hold down the **LMB** while using the **Mouse Wheel** to adjust the radius of the influence. On Mac, hold the "fnKey" down and hit "page-up" or "page-down"). In 2.41 you must 'grab' the vertex first - only then can you alter the sphere of influence (in my version, 2.42a, the 3D cursor had to be snapped to the selection before the wheel appeared).

Noob Note: To change the radius on a Macbook press "fn"+ up/down. OSX uses function + arrow keys as replacements for "home", "end", "page up" and "page down" keys.

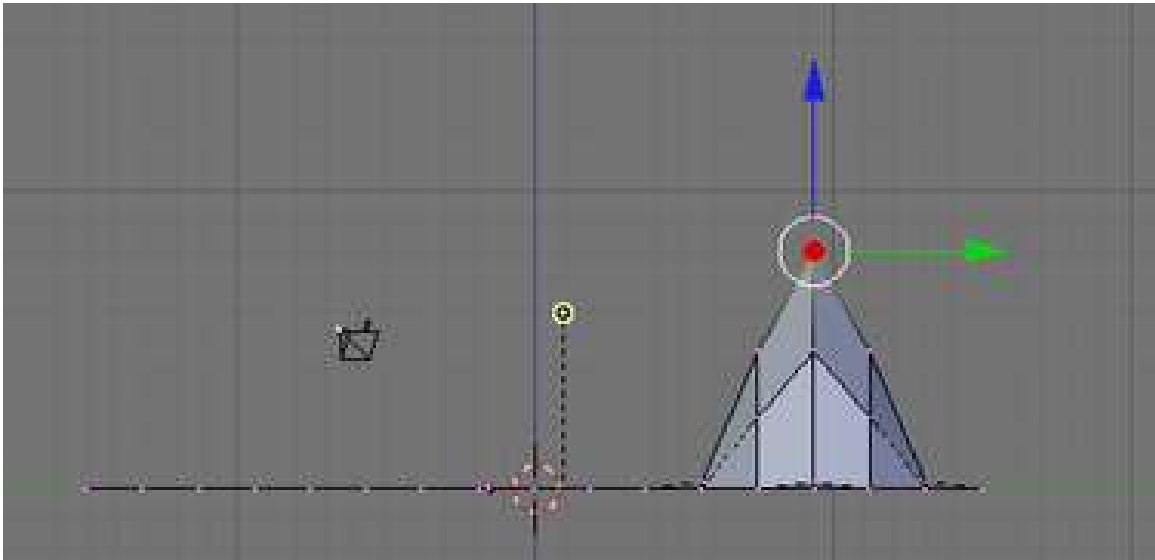


Figure 187

- Move the vertex up about 8 units on the Z-Axis. Do this by dragging the cursor up a little, and press the **MMB**; this should restrain the movements along the Z-axis. Now use **CTRL** to move it precisely. Alternatively you can use **ZKEY** to restrain movements to the Z-Axis and type **8** and hit **ENTER**. In older versions of Blender you may need to hit the **NKEY** before typing the number 8.

Congratulations, we just created our first mountain. Now it's time to see what other things we can accomplish with the proportional editing tool.

44.3. Peaks vs. hills

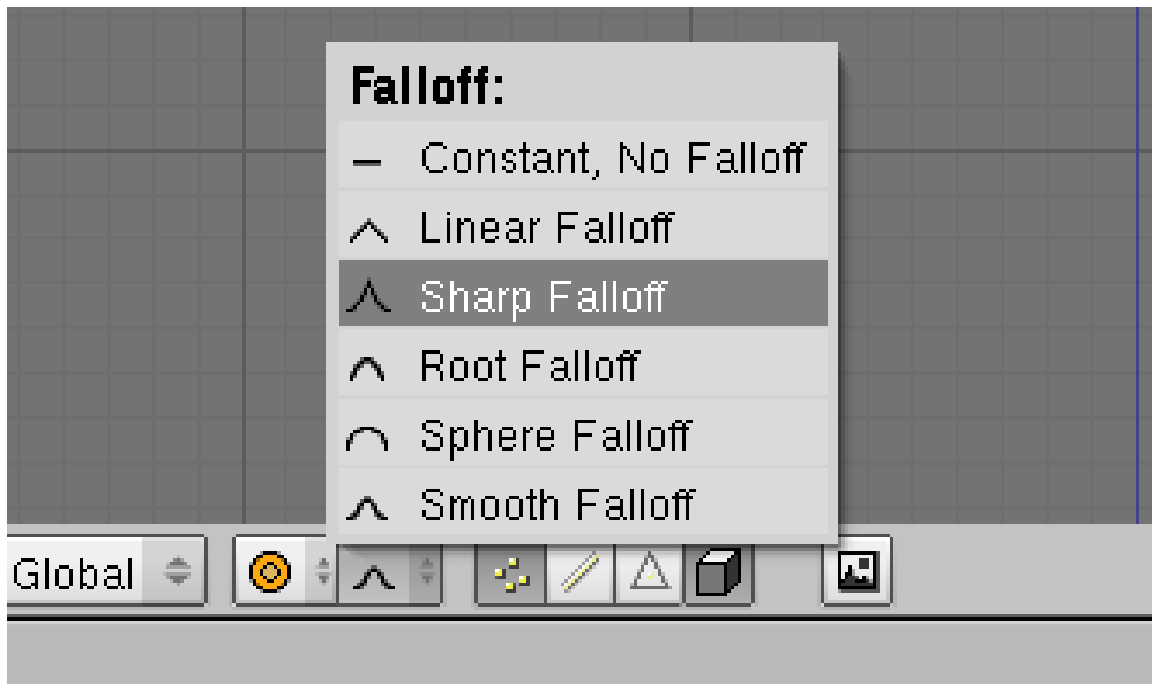


Figure 188

The 2.37 and onward releases offer at least 6 types and 2 modes of proportional editing. The previous release only has 2 of these types: Smooth and Sharp Falloff. We'll take a look at the difference between these two now.

- Change to top view again with **NUM7**. You'll notice that now your "mountain" looks like a few differently shaded squares in the grid; you're looking down on shaded tiles, but in the Z axis, they're all still perfectly aligned with the original grid.
- Select another vertex away from the first. Let's say 4 from the bottom 4 from the right (counting the vertices on the edges).
- Change back to the side view with **NUM3**
- Select Sharp Falloff from the menu on the bar of the 3D View. Alternatively, using **SHIFT+OKEY** will switch from one to the next of the 6 proportional editing modes while using the Proportional editing tool.
- As before, move the vertex up 8 units on the Z-Axis (*Note: The radius of influence will still be the same size as when we last used it*).
 - **GKEY**
 - **ZKEY**
 - Type **8** and hit **ENTER**

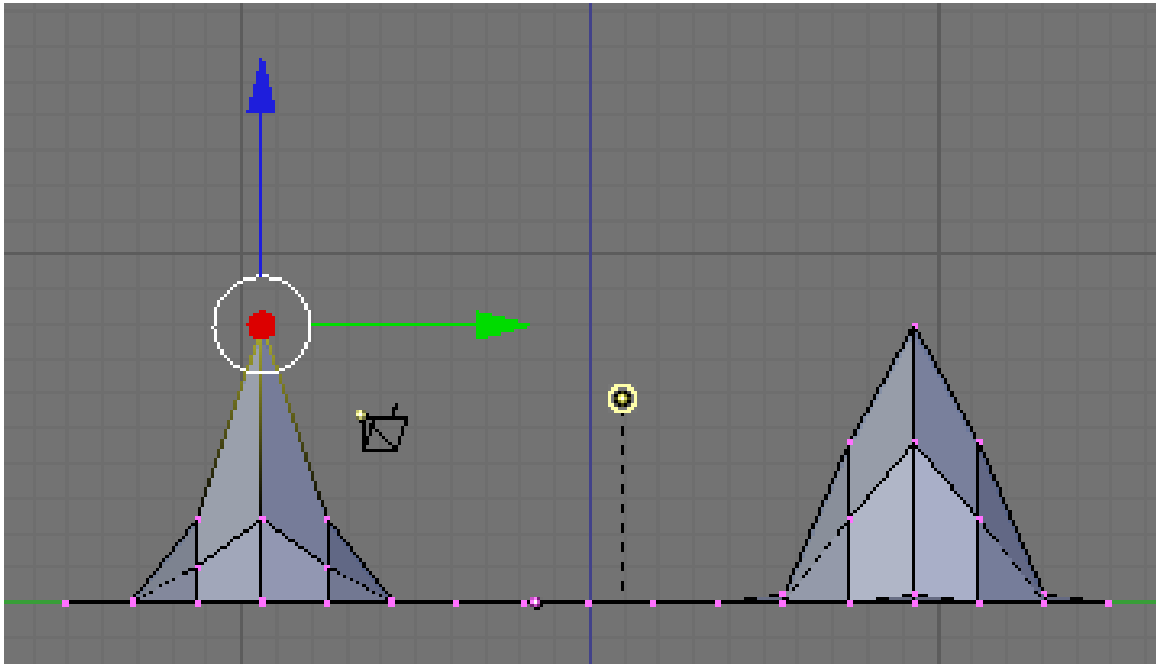


Figure 189

Now we can see the differences between the sharp and smooth falloff. The same number of vertices are affected in both cases; only the degree to which they are affected is different.

The different proportional editing modes can be selected from the box immediately to the left of the proportional editing type box. The mode box contains three options: Off, On, and Connected. "Off" means that proportional editing will not be used. "Connected" means that only vertices linked to the selected vertices will be affected by the radius of influence. "On" means that all vertices will be affected.

44.4. Shaping the world

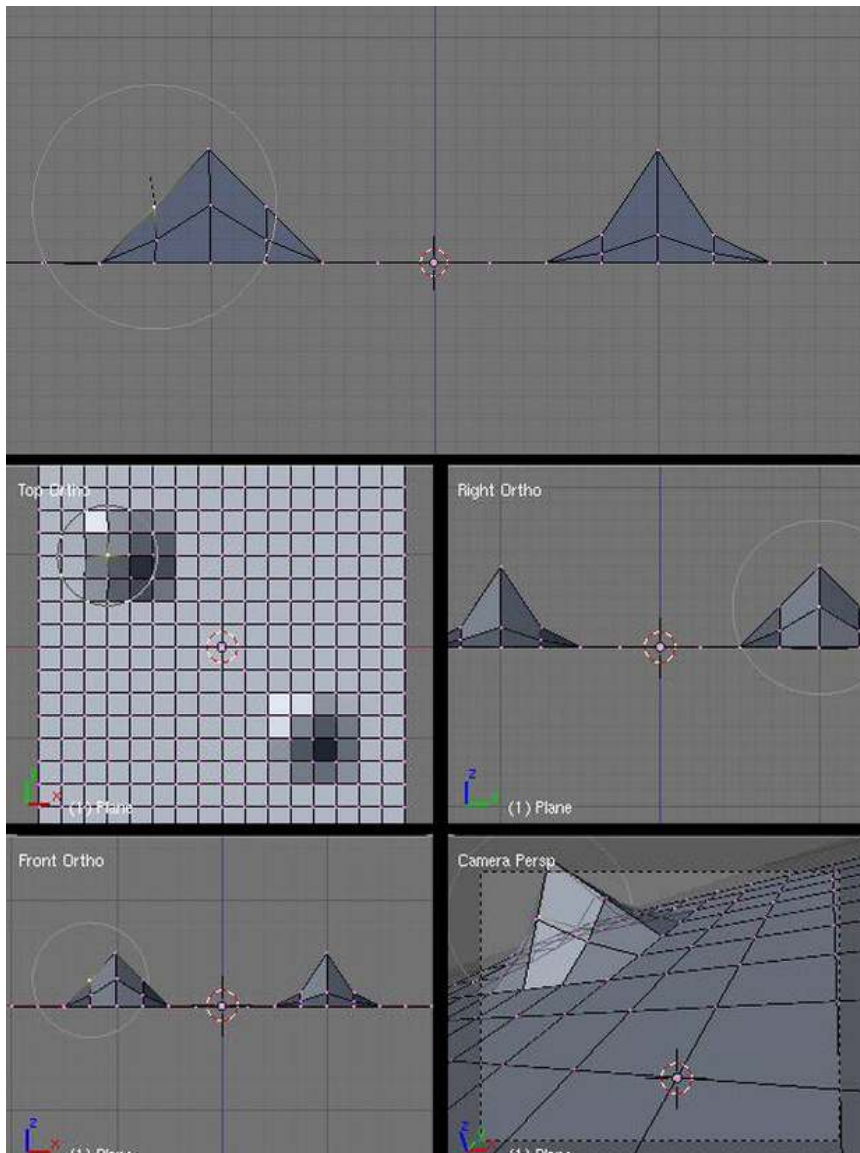


Figure 190

Now that we've created a couple of Mountains, it's time to see how we can use proportional editing to shape them.

- First make sure we're in side view (**NUM3**).
- Then on the smooth falloff mountain, the first one we created, select the vertex that is immediately down and left from the topmost point.
- Press **RKEY** to rotate, scroll the **MMB** to change effective radius so it includes other points. Your screen should look like the photo to the right.

You can see the size of the proportional editing circle, and that there is only one vertex on the mountainside selected.

- Next hold **CTRL** and rotate everything by **-90**. Alternatively, use **RKEY**, **NKEY**, and type **-90** and press **ENTER**. Your mountain should now look like this:

Noob note: be careful about the range of affected vertices. If the range is too small, then rotating will affect just the selected vertex. If the range is too large, it will rotate everything together. You can adjust the range by using the scroll wheel on the mouse.

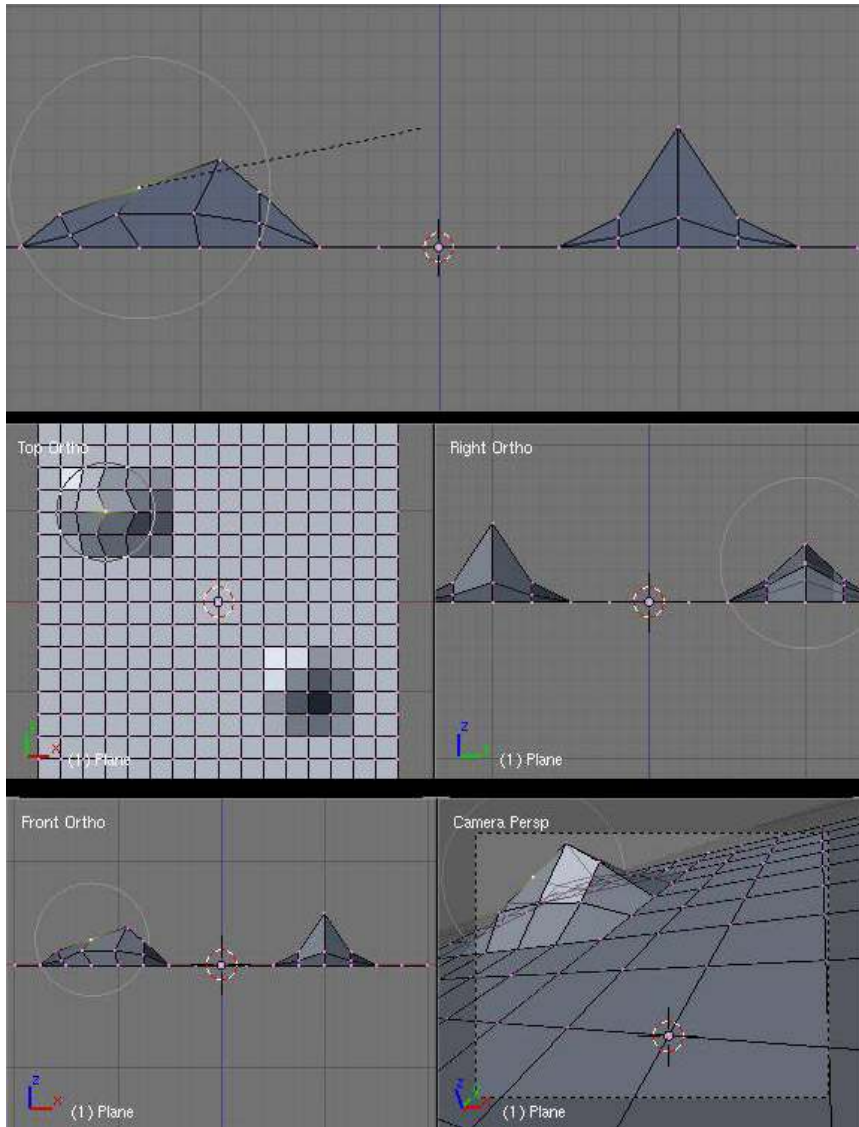


Figure 191

Notice that the vertex itself did not move; since it is at the center of the circle it had no effect. The adjoining vertices within the edit circle were rotated around it in decreasing amounts the further from the center they are. Try doing it again with a larger proportional editing circle. Feel free to play around with scaling or rotating from different view points (don't forget that you can also use **GKEY** to move vertices vertically or horizontally).

Try viewing your world from top view while rotating with a large effective radius. You will see the nearby vertices move close to the full amount while vertices further away move less.

44.5. Smoothing things out

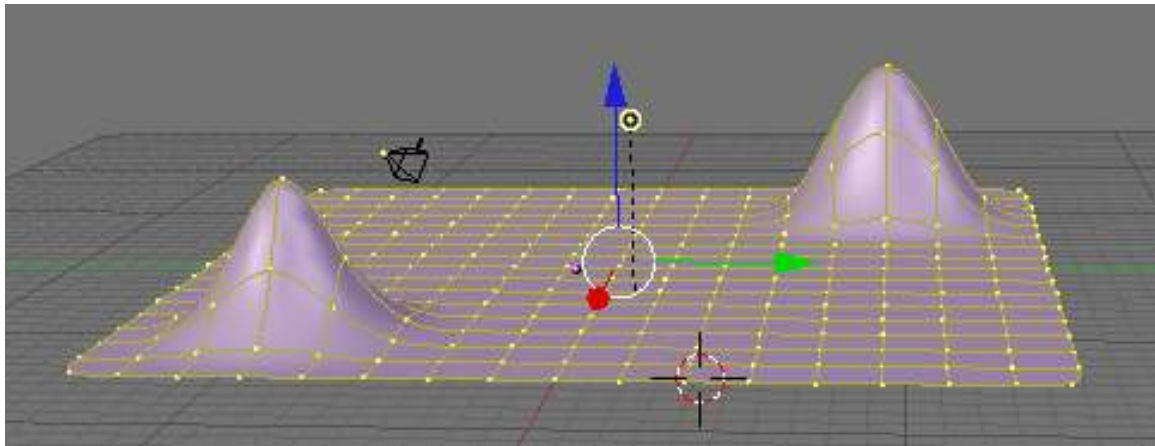


Figure 192

Now that we have a couple of budding mountains, you probably think they look kind of choppy. Sure they would be good if we were making an 8-bit console game, but we're working with 3D here, we want things to look sharper (or maybe smoother) than that. There are a couple of approaches to this. The first is to use more vertices when we create our plane. And I won't lie, it works. But it's also a HUGE resource hog. It would take your home computer hours of work just to keep things updated, let alone run it. So instead, we fake it. The easiest way to do this is to turn on *SubSurfaces* (we saw this in *Detailing Your Simple Person 1*¹.) For our purposes, let's set the subdivision (*Levels*) to 2. Also, ensure our SubSurf algorithm is set to *Catmull-Clark* (this is the default setting).

Now, you'll notice that with SubSurf on, we lose a lot of hard edges that we had, essentially we have no sharp corners any more. I don't know about you, but to me that doesn't make for a very interesting mountain range. So to restore our corners, we are going to use *Weighted Creases for Subsurfs*.

- First turn off proportional editing with the **OKEY**, and ensure we're in side view with **NUM3**
 - *Noob note: That means, the letter O key, not the zero key. Also, NUM3 means "3 on the numeric keypad," not on the top row of the keyboard.*
- Next, while still in edit mode, change to *Edge Select* mode with **CTRL+TAB** and select *Edges*. Alternatively press *Edge Select Mode* button at the bottom of the object window.
- Under the *Edit buttons* under *Mesh Tools 1* (*Mesh Tools More* in versions 2.46 and later) ensure that *Draw Creases* is selected (**2.58** - By default on the left hand side you'll find the tab "*Mesh Options*" and a dropdown called *Edge Select Mode*. Here you'll find *Tag Crease*). (*Mesh Tools 1* may be off the screen, if so, use the scroll wheel when over the Edit Panel to reveal it. Alternatively, you can pan the buttons window by dragging with the **MMB**; or you can collapse some of the panels by clicking the arrow next to their names.)
 - *Noob note: "Mesh Tools 1" (also known as "Mesh Tools More" in 249.2) is a separate panel from "Mesh Tools" and from "Mesh". By moving the mouse cursor over the buttons window (remember this 'activates' windows) and using the mouse wheel, you will cause the tabs to scroll from side to side, revealing some you couldn't see before. If the one you need still isn't there you*

¹ Chapter 31 on page 183

should check to make sure you're still in edit mode, or make sure it isn't 'minimized' which is done by clicking the little white arrow at the top left of the tab.

- On our Sharp Falloff mountain, the second one we did, select the two edges on the right. (see image below)

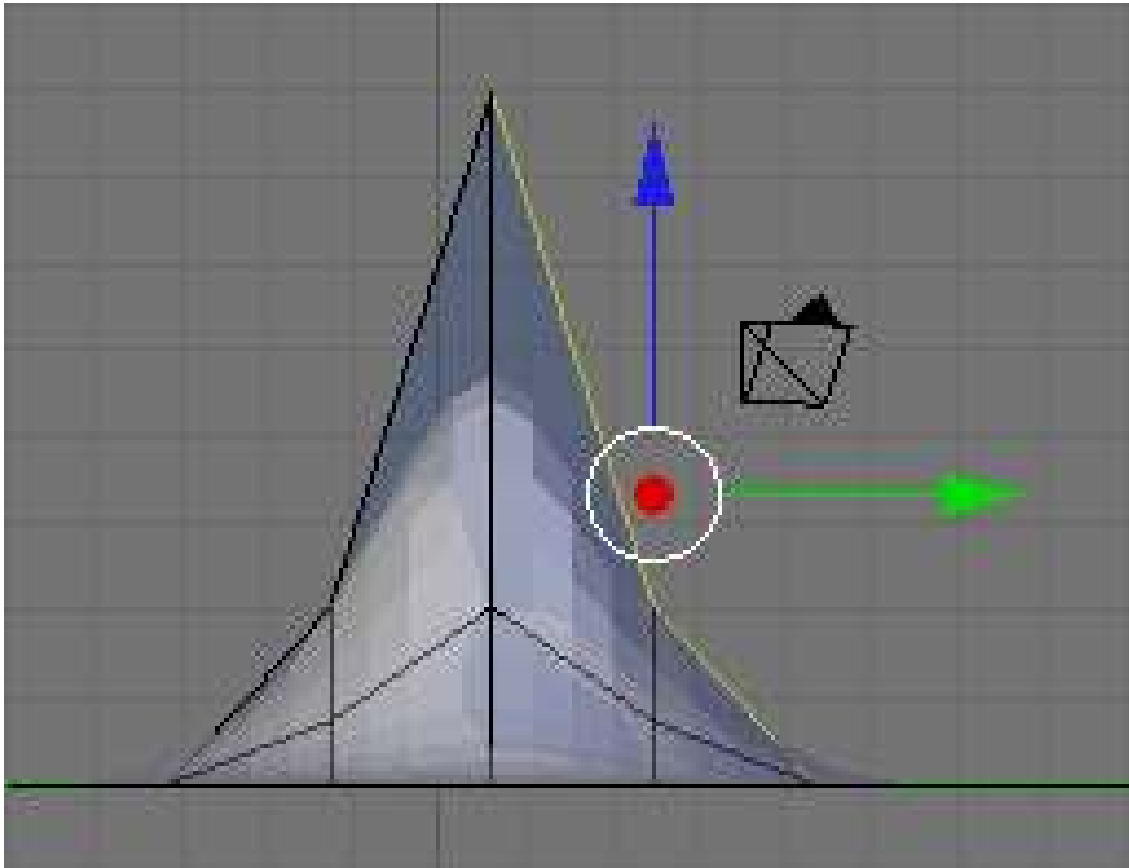


Figure 193

- Press **SHIFT+EKEY** or **SPACE** → *Edit* → *Edges* → *Crease SubSurf*, then move the mouse away from the edge until the edge *Crease* reads 1.000 in the 3D viewport header. If moving the cursor there seems to be impossible, just hit 1 and enter.

As you move the cursor away from the edge you will notice two things. The first is that the edge becomes thicker as we move from it; this is showing how much of a crease we have (with *Draw Creases* turned on). The second is that you will notice the subsurfed mesh moving closer to the edge as the sharpness increases.

44.6. Naturalness

Press **CTRL+TAB** and select vertices. Then go into front view **NUM1**. Select the second vertex from the top in the centre of our Sharp Falloff mountain, then go into side view **NUM3**. Push **GKEY** and drag the vertex inwards, not too far or your mountain will come out of itself on the other side. Just bring it in enough to make a small indent.

Then grab the top vertex and pull it down a small amount. You will notice that there is a small "crunch" in your mountain.

Don't forget to select all, press **W**Key and hit the **set smooth** button to smooth everything out.

OK, so your mountains are starting to shape up. But they still look a bit too neat. You could spend time moving each individual vertex but the chances are your model will still lack the natural feel. What we need is some chaos. Thankfully this is quite easy to accomplish. Firstly select the vertices that make up your mountains, all of them and a few around the base (box and circle select will make this easier). Select a few vertices between the mountains too. Next we use something called fractals. Fractals are chaotically (i.e. randomly) generated variables. In short you can use these variables to give your mountains a "wobbly" look.

Figure 194 Fractal option in 2.6.x

Fractals are located in the *Mesh Tools* section of your edit buttons (next to Noise, Hash and Xsort) or in the options for subdividing labeled *Fractal* under *smoothness* (Access it through the specials menu (**W**KEY) and selecting *subdivide 2.61*) Click it and you'll be asked for a value. This value is the strength of the fractal. 1 is very low and will barely change your model. 100 is very high and will twist your models into very odd shapes indeed. Have a play with different values until you find one that you like. Around about 15-30 should do it. Hit OK and presto, your mountains have been transformed from clinical neatness, to lumpy chaos.

- If you make too many fractals, your computer will slow down. However, the more fractal you add, the more bumpy and realistic it looks!

*Noob Note: Repeatedly using the fractal tool seems to rapidly multiply the amount of vertices on your canvas. I suggest using the tool once, and if the result isn't satisfying, undo the result (**CTRL + Z**) and try it again with a different fractal strength. Even after undo, your selected vertices remain selected.*

Noob Note #2: If you have a new enough version of Blender (2.49 and up definitely has it, not sure about lower versions), there is a setting called Random in the Proportional Editing settings. Using this with a high radius may give a desired result for you without adding more mesh.

44.7. Adding your guy with the hat

If you changed to a new layer-- press (**SHIFT**) select the layer with your guy on it or (**SHIFT + 1**). This will display both layers. If you would like to move the landscape from the second layer to the first then with the landscape selected, press (**M**), and select the layer(s) you would like it on. note-make sure you're in object mode. If you started a new project-- To insert your guy with a hat, you can simply go to **file > append or link**(shift+F1).

- Then select the file from the previous tutorial.
- You will then see a list of objects you can insert or 'append'.
- First go into the 'Object' folder. Unless you renamed him/her, your person will probably be named "cube." Select it.
- Make sure that 'Append' button is pressed; otherwise you will not be able to scale or translate, or edit your model at all. (The 'Link' button will link a copy of your object into the current scene, and

will update any changes when you reload the file. Unfortunately, this includes location and size, so we will not use it right now.)

- Press 'Load Library' to place your guy into the mountain scene. Please note, the *Load Library* button is top right. Confusingly, there is some text bottom left that says *Load Library* as well, but isn't clickable.
- Repeat the process to get the hat (probably named "circle"). To reset the parental relationship, see the instructions at the end of the *./Putting Hat on Person*² tutorial. To scale them to the appropriate size, go to object mode by pressing **TAB** (unless of course you are already there). Select the object you want to resize, then press the **SKEY** and use the cursor to scale as you see fit. To undo any mistakes, use **CTRL+Z**.


Noob Note: To import the file once you have the list select the item, i.e. cube, and in the bottom right select "Active Layer" that will append it to your current proj when you click "Load Library" your work should be visible now.


Noob Note: Multiple items may be appended simultaneously. For instance: both 'cube' and 'circle' objects can be *appended* at the same time by highlighting both object names within the file browser by means of a RMB Click. RMB on object again to deselect.

45. Modeling a volcano

In this module, you will create a volcano using the proportional edit fall-off tool. You should be comfortable with deleting and adding meshes.


45.1. Adding a Plane

Delete the basic cube. Add a plane, and  scale it up by 10. Rotate it so you see it in top-view (make sure its in Orthographic view too).

Enter Edit mode and subdivide (with ) 5 or 6 times. More subdividing will give you a "smoother" volcano, but it also needs more CPU power.

"Subdivide" divides every square in the plane into four new squares. So every time you press "Subdivide" you will have four times as many squares as before. "Subdivide Multi" will make x horizontal and x vertical lines through your existing squares, so the new number of squares is: $(\text{squares_old}) \cdot (x+1)^2$, whereas x is the number you enter.

45.2. Making the Mountain

In top view, select one of the points in the middle of the plane. With this point selected change to side view. Press the , which enables the "Proportional Edit Falloff" tool in the Menu-Panel beneath the 3-D-Window. As seen in the previous tutorial Blender 3D: Noob to Pro/Mountains Out Of Molehills¹ when you move a vertex while edit falloff is enabled, all vertices in a defined radius of the selected vertex will align with the selected vertex when its position is altered. How they are adjusted can be chosen in the tab on the right of the yellow dot. I propose using "smooth falloff".

1 Chapter 44 on page 263

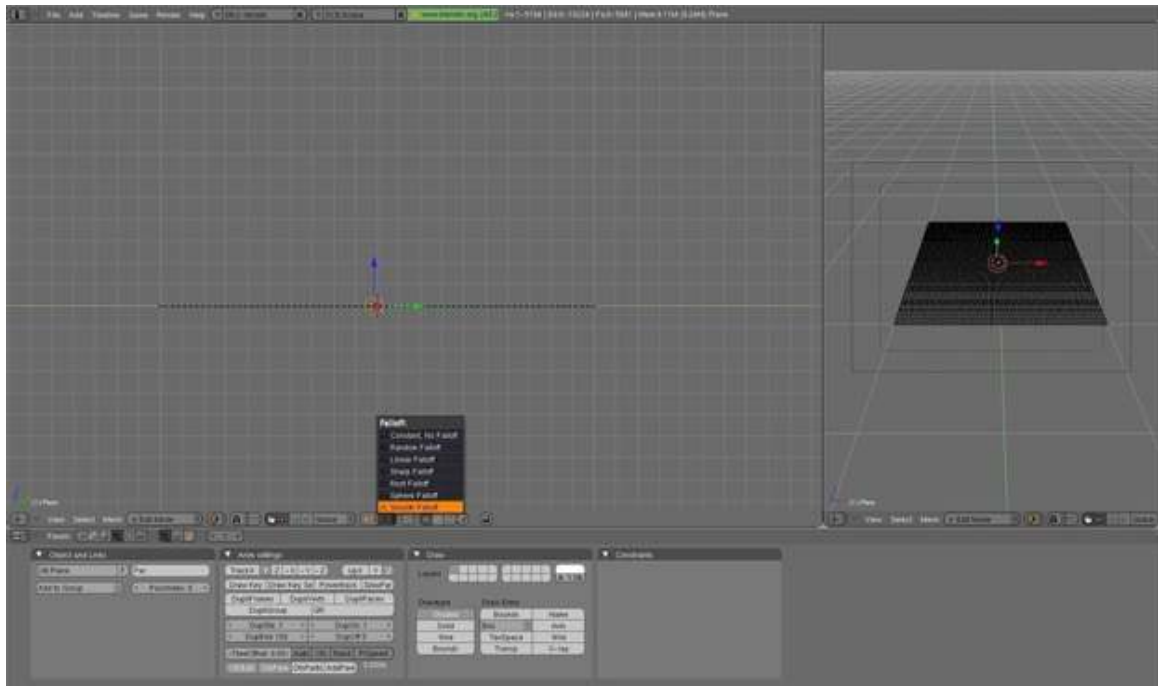



Figure 195

Now grab the vertex with . You will now see a gray circle. You can change its size with the mouse wheel. Every vertex inside this radius will be affected by the falloff. Change the size of the circle so almost the whole plane is in it.

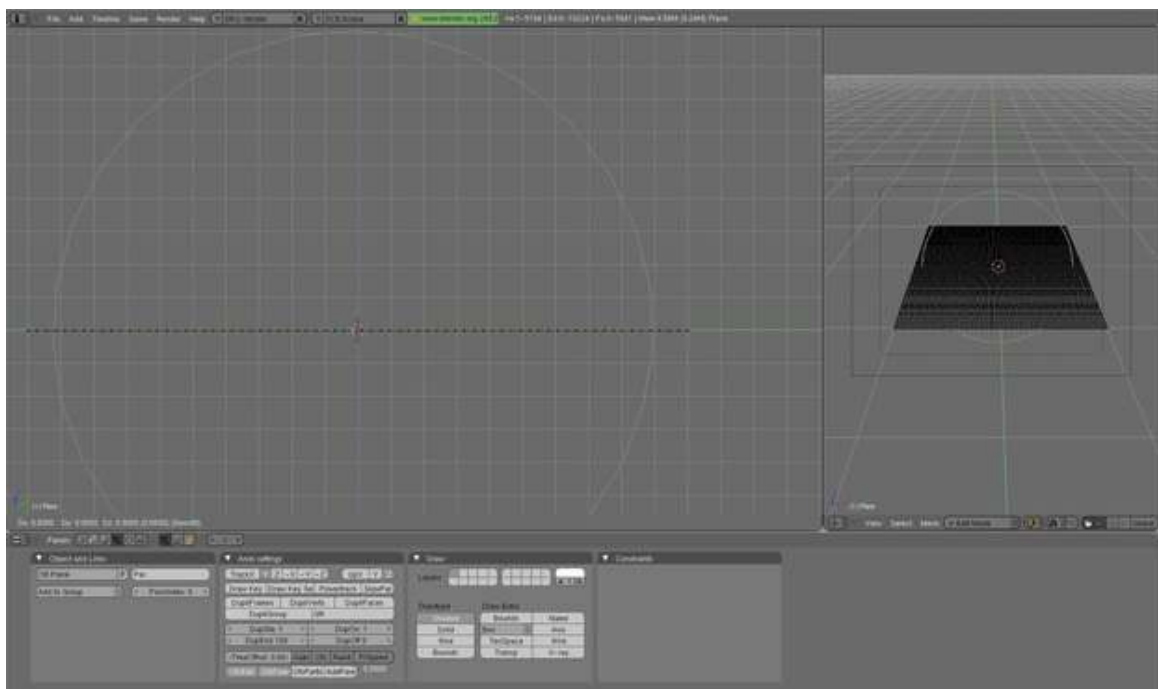


Figure 196

Now move the vertex a bit upwards, as seen in the picture. Optionally you can lock the z-axis to make the volcano go straight up by pressing **Z**.

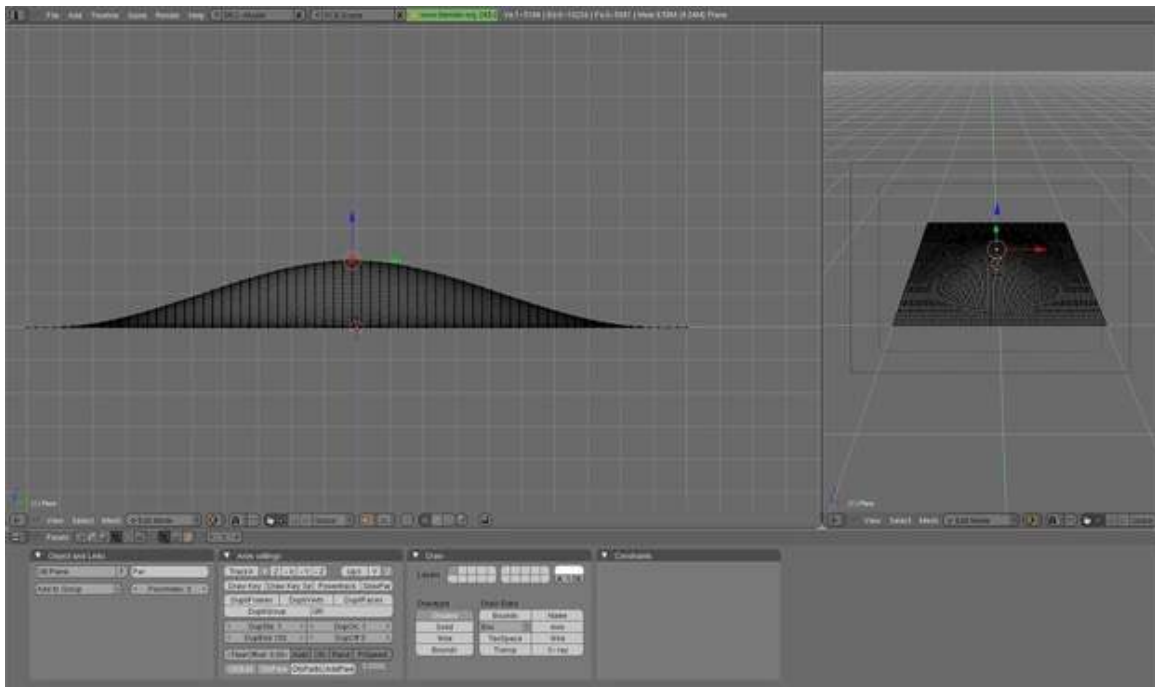


Figure 197

As you can see all the other vertices will shift upward. We could keep moving this vertex at the same rate, but that would cause the plane itself to rise and bend, and that's not very good. So press **LMB** to apply the changes, grab the same vertex a second time and repeat the previous exercise as before, except now choose a smaller radius for the circle (**G** → **Z** → scroll **MMB**).

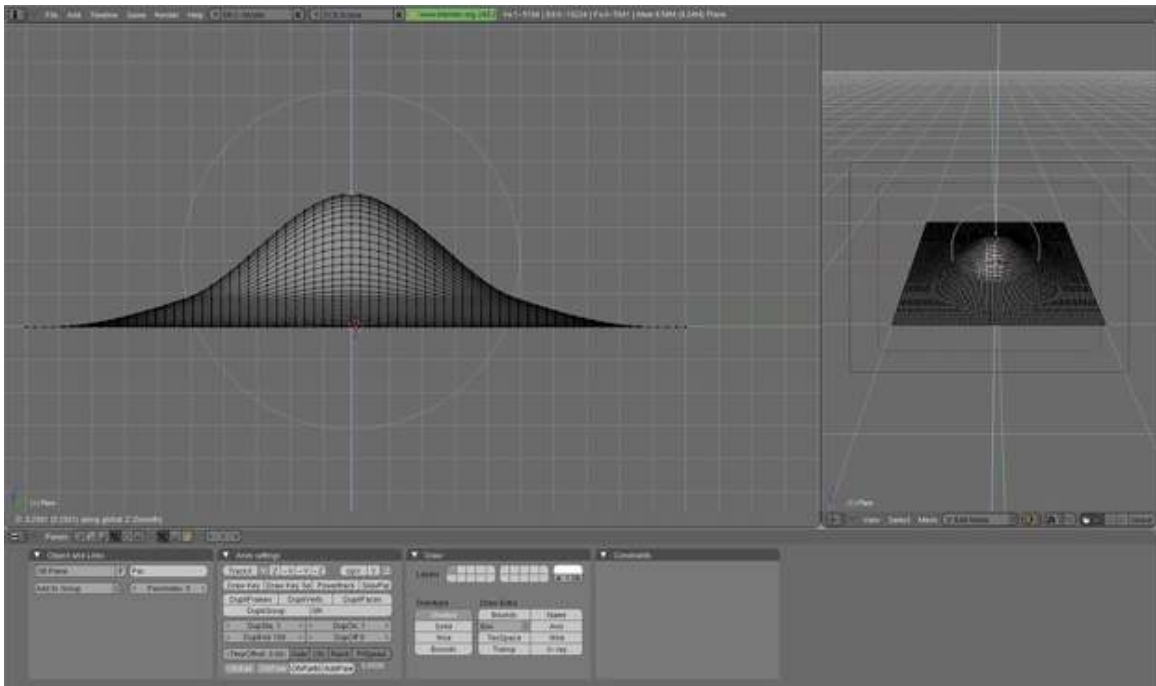


Figure 198

Repeat this two or three more times and you will get something like this:

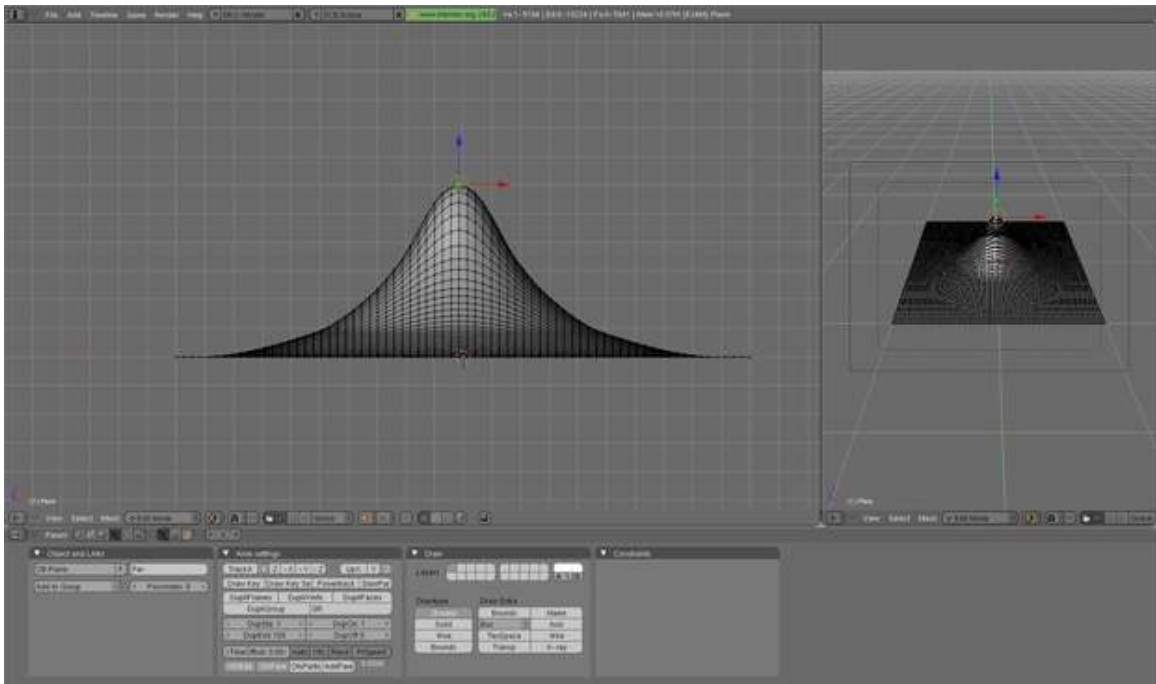


Figure 199

45.3. Forming the Crater

Now we're going to create the "hole" on the volcano. First change the falloff to "root". Grab the vertex one more time, change the size of the circle so it's more or less as seen in the picture.

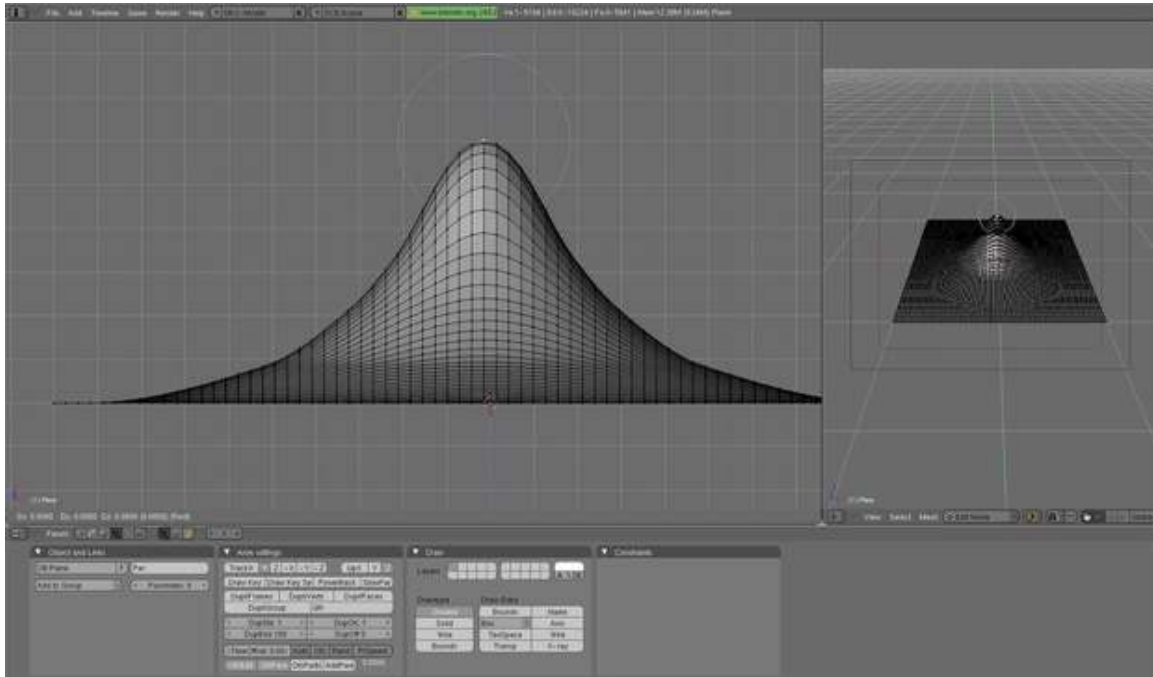


Figure 200

Grab this vertex down a bit, apply, grab it one more time with a smaller circle. You now should have something like this:

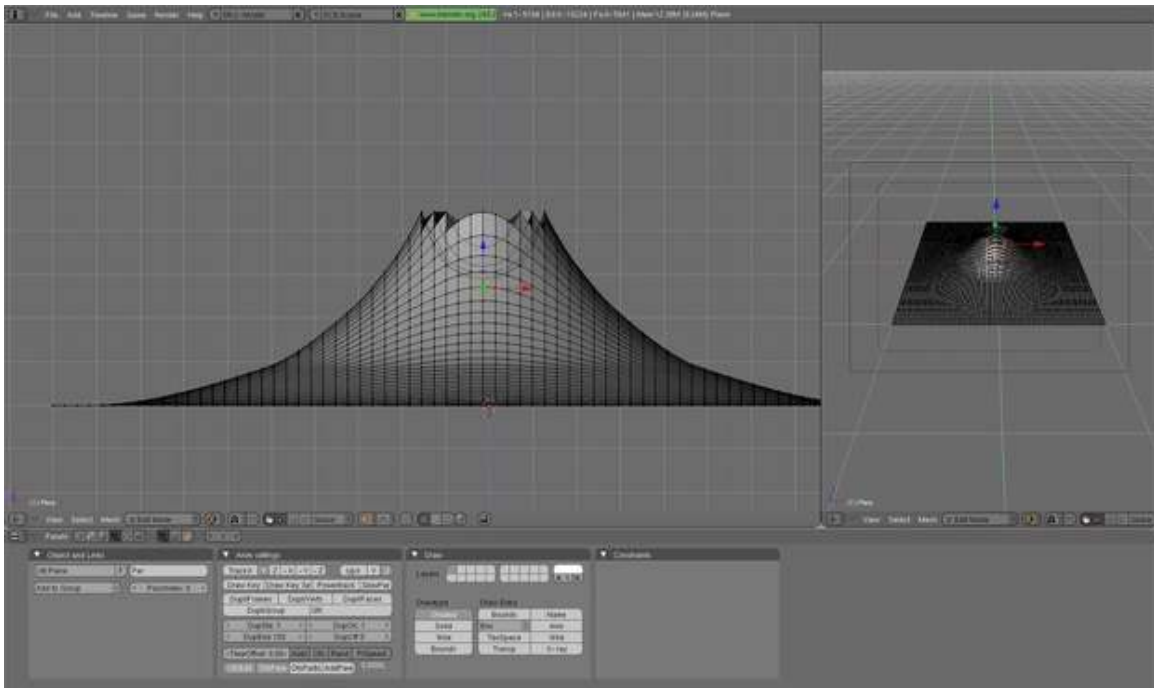




Figure 201

If yours has such a jagged border like mine, you should follow the following steps: First disable the fallout tool with . Then select a vertex and grab it, then pull it up so it's on the same height as the other. Do this for every vertex till they're more or less the same height.

You can also select the jagged vertices and the vertices around them then press  → Smooth (Older versions: **Space** → edit → vertices → smooth) and this will get rid of some of the jagged edges. (I suggest using this method since this is only a basic volcano, and using Subsurf after this will make the jagged edges all but invisible anyway.)

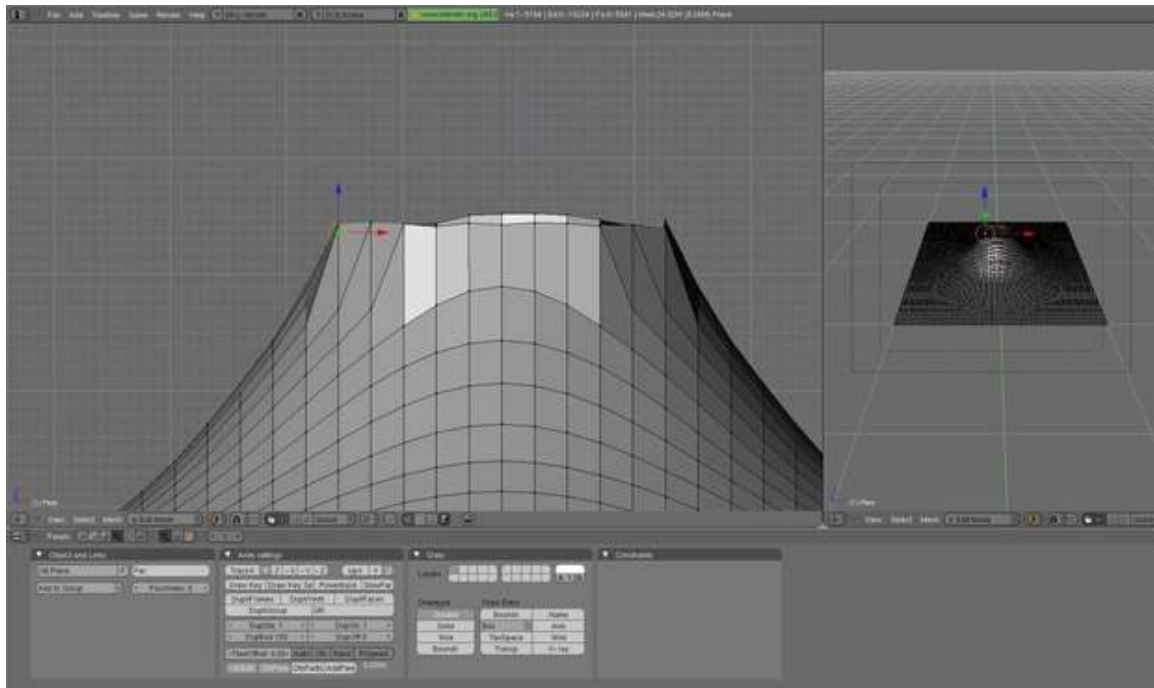


Figure 202

45.4. Subsurf

Now **TAB** to get into **Object mode**. Add a Sub-Surf Modifier to the Mesh, set at level two ('**Modifiers**' tab → Add Modifier → Subsurf → Catmull-Clark → Levels: 2; Render Levels: 2).

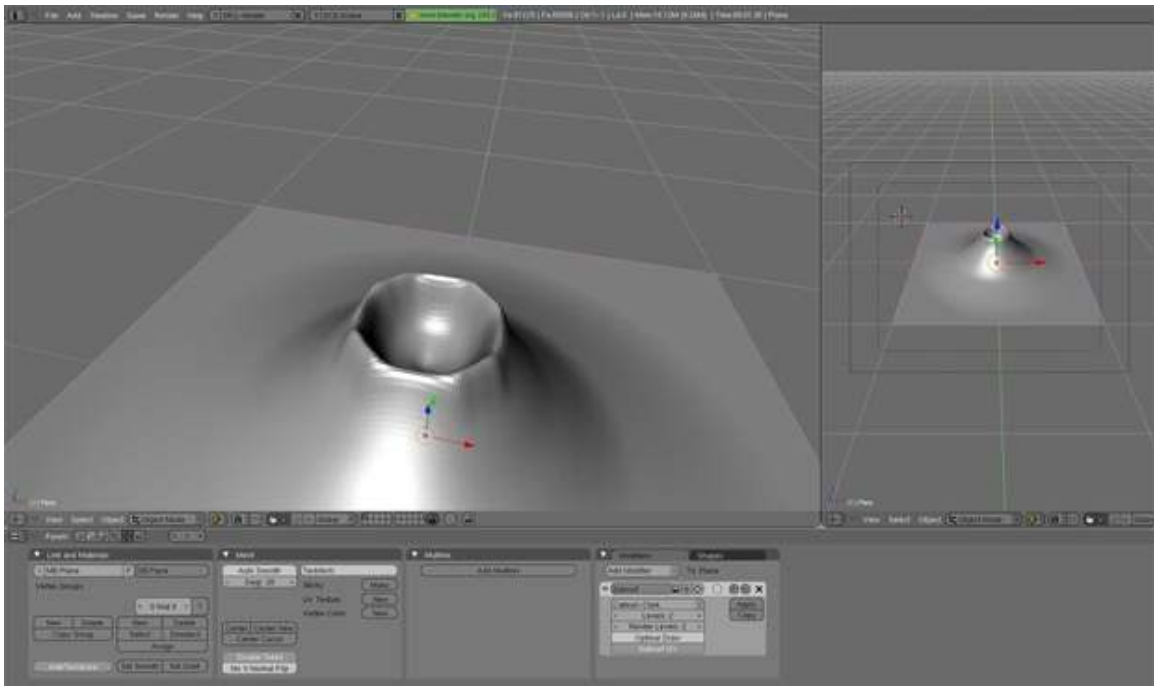


Figure 203

Do not apply these settings yet, first we'll do a test-render. Press **F12** to enter Render, after adjusting the camera. If you like your volcano you may apply the modifier, but if you have some rather nasty edges like I do, don't apply the modifier yet. We'll correct this now.

First click on the marked button to disable the subsurf-modifier in Edit-mode.

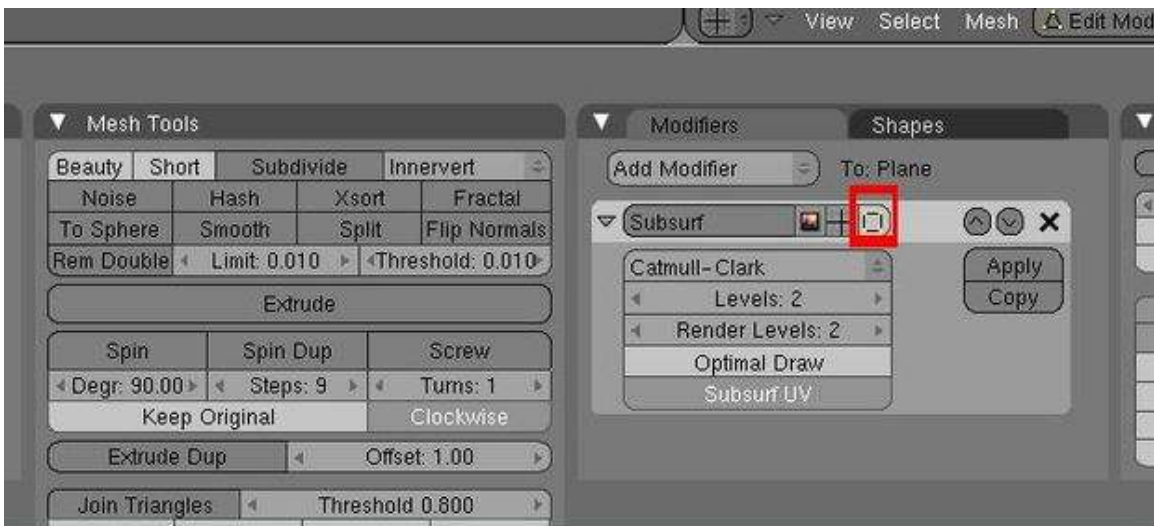


Figure 204

Then enter Edit-mode. Then select the vertices on the top and pull one after another a little bit outwards, so they form a nice circle.

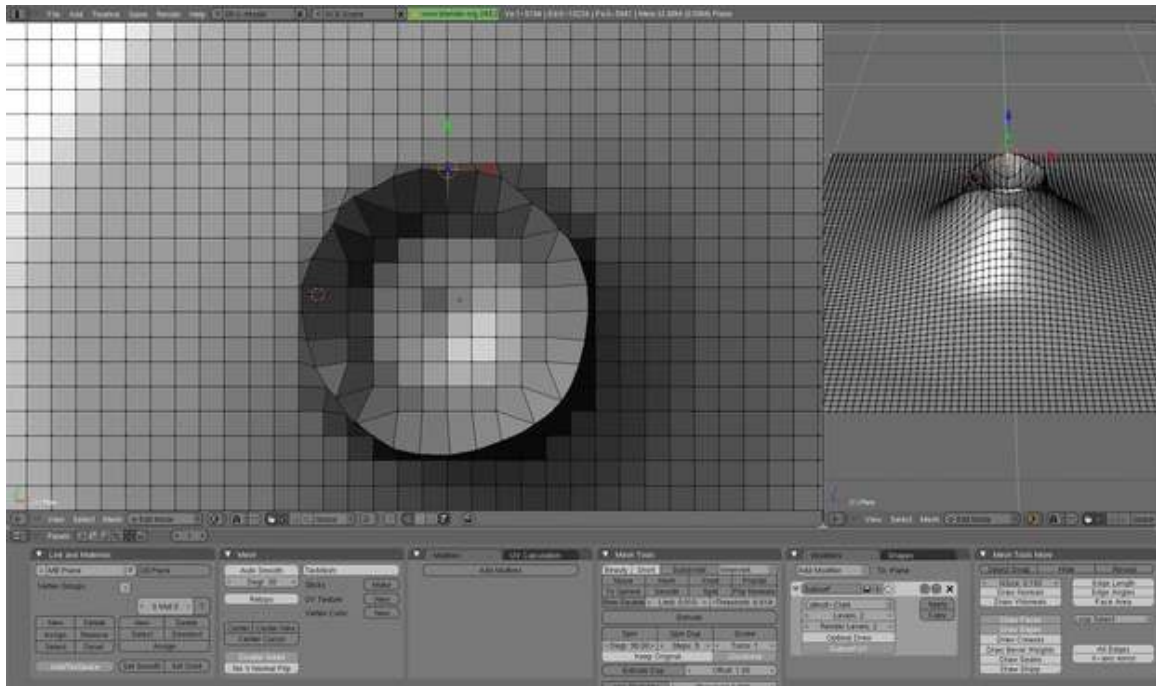


Figure 205

Once you have done this exit Edit-mode and press the "Set Smooth" Button in the Link and Materials Button. If you Render now again, it should look a lot better. (Don't worry about the color)

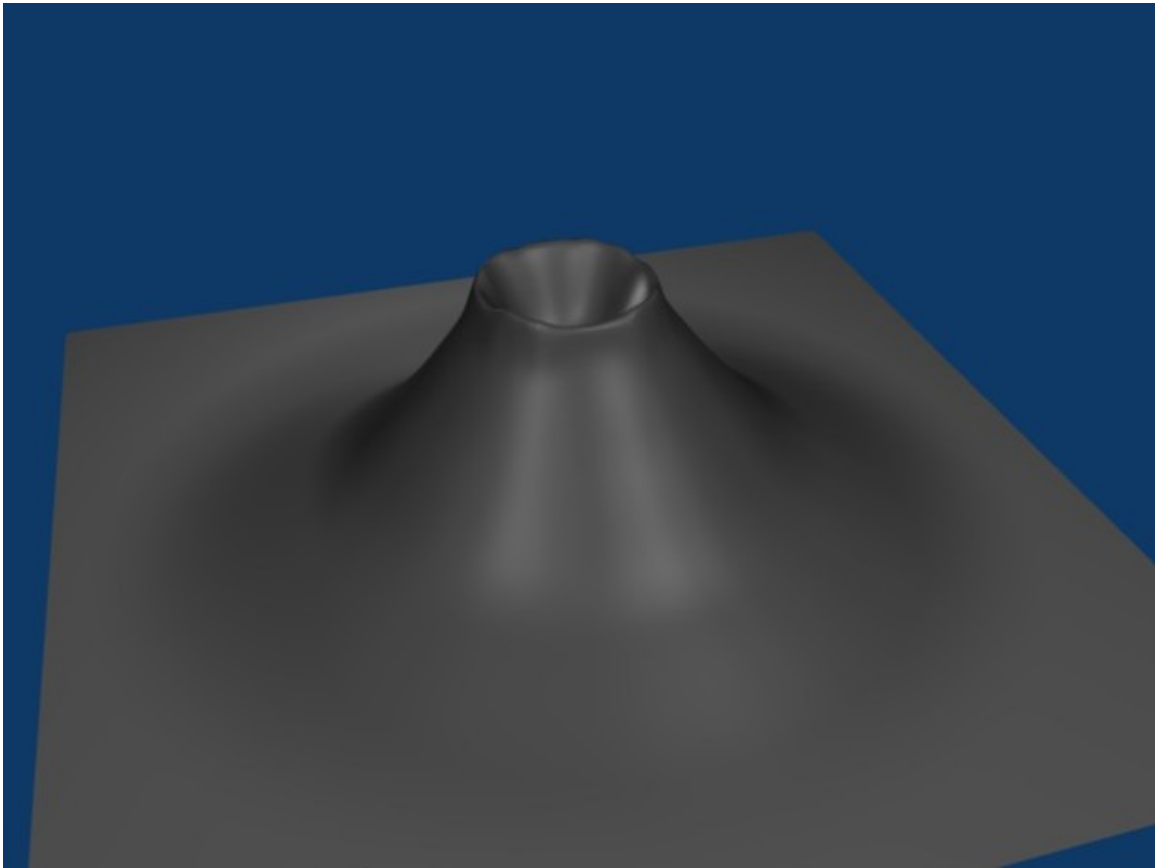


Figure 206

45.5. Adding Magma

Let's add some "magma" using lighting.

1. Press **[SHIFT] + [S]** and choose *Cursor to Center*.
 2. Press **[SHIFT] + [A]** and choose *Lamp → Point*.
 3. **Older versions only:** press **F5** to enter the shading menu
 4. Change the color to reddish-orange. (Red: 1, Green: 0.1, Blue: 0)
 5. Set the Energy to around 7.
 6. Raise the light until it's just above the bottom of the crater (**[G]**rab along the **[Z]** axis).
 7. If the ground level of your plane is reflecting light from the lava lamp this is because the bottom of your crater is above ground level of the plane you created; you'll need to turn on ray-tracing. in the object data menu for the light, open the Shadow menu and click "Ray Shadow"
- Alternate 1: Spot Lamp
 1. Change the light's type to *Spot*.
 2. Raise the light until it's covering most of the crater. If the light is not pointing down, **[R]**otate and angle it downwards.
 - Alternate 2: Area Lamp

1. Change the light's type to *Area*.
2. **R**otate along the **Y** axis: 180 degrees.
3. Set Gamma to 2.
4. Set Distance to around 5.

Experiment with the values and positioning to get something that works with your volcano.

It should now look like this:

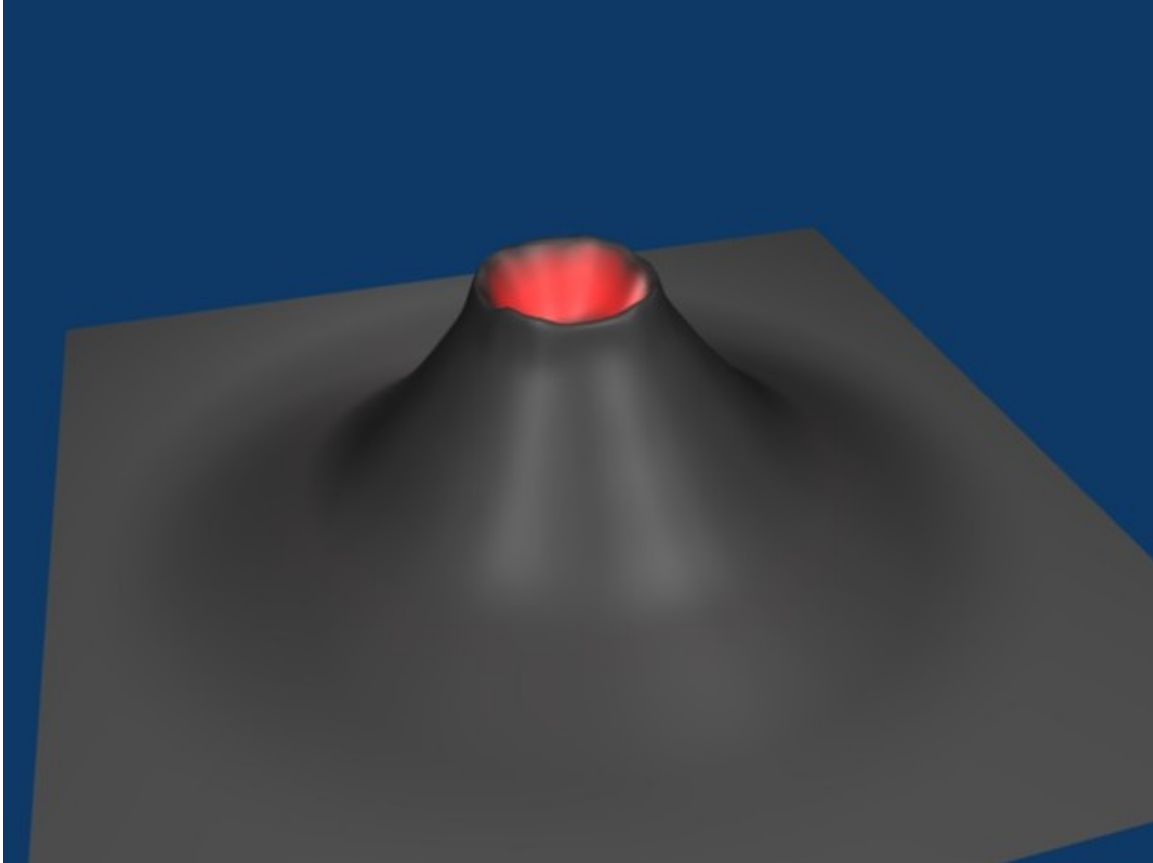


Figure 207

45.6. Varying the Terrain

Next, let's set the volcano's material.

1. **RMB** on the volcano plane.
2. Select the "Material" button



Figure 208

and press New.

3. Change the color to ashen gray. (Red: .260, Green: 0.230, Blue: 0.230)
4. Select the "Texture" button and press New.
5. Change the Type to Stucci.
6. In the Influence panel, uncheck Color, and check Normal. Set the Normal slider to 0.5. This will render the texture as a bump-map.

Older versions:

Select the volcano and press F5. Keep Pressing F5 until the Materials Buttons (symbolized by a red ball) is highlighted. Then add a new material. You do this by clicking the Add New button in the Links and Pipeline Panel. Once you've done that, set the settings similar to the picture below. Now press F6, then add a new texture to the material. Choose a stucci texture, set the noise size to 0.15. Now switch back to the materials-window (F5) and click on the "map to" tab. Deselect the "col" button and select the "nor" button. This will render the texture as a bump-map on the volcano. Set the "nor slider" to 0.5, which should be the default. Switch to the "map input" tab and choose "tube".



Figure 209

If you now render you should get something like this:

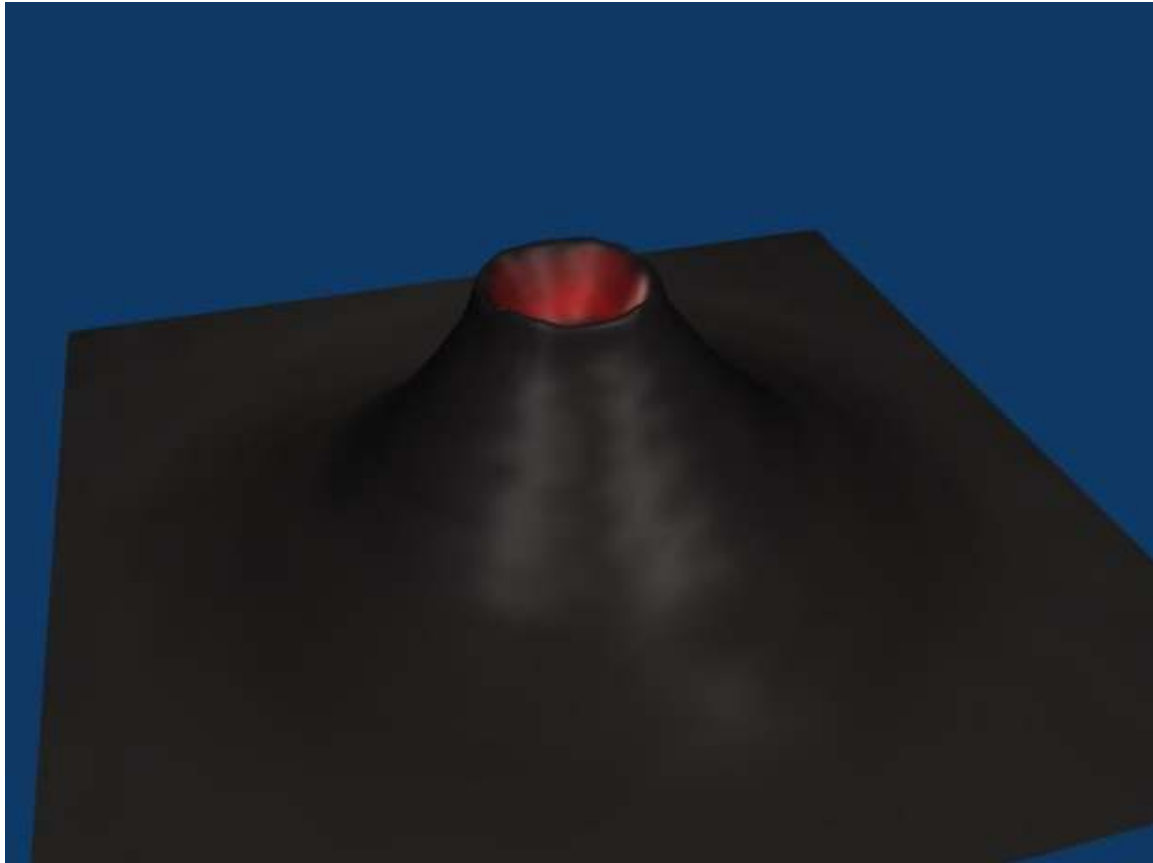





Figure 210

This looks really smooth, like clay pottery. To get a more rough-looking volcano, try out these options:

- Option 1: Subdivide and increase the fractal.
- Option 2: Decrease the texture's basis size (in the "Stucci" panel when you select the "Texture" button).
- Option 3: Proportional edit tool.
 1. Press  to turn on Proportional Editing mode. Select Random fallout.
 2. rab the center vertice, and raise it along the  axis.


Older versions:

Go into Edit mode, select all Vertices, and use the fractal (set from 15 to 30) to really get things looking rocky and mountainous. **TAB** (Edit mode) → **F9** → **AKEY** to select all → Mesh Tools → Fractal → 15 - 30 (15-low, 30-high) → OK → **TAB** (Object mode) In Blender 2.5 you can use the random proportional edit tool: use **NUM7** to switch into top view, grab the central vertice of your volcano using a large-radius random proportional edit, and pul it slightly into Z-direction.

(A note: Seems there is no need in subsurf at all since fractal tool will dramatically increase vertices quantity.)

46. Penguins from spheres

46.1. Setup

Start with the default scene¹: it should contain a selected cube. Delete this cube by pressing  → *Erase selected Object(s)*.

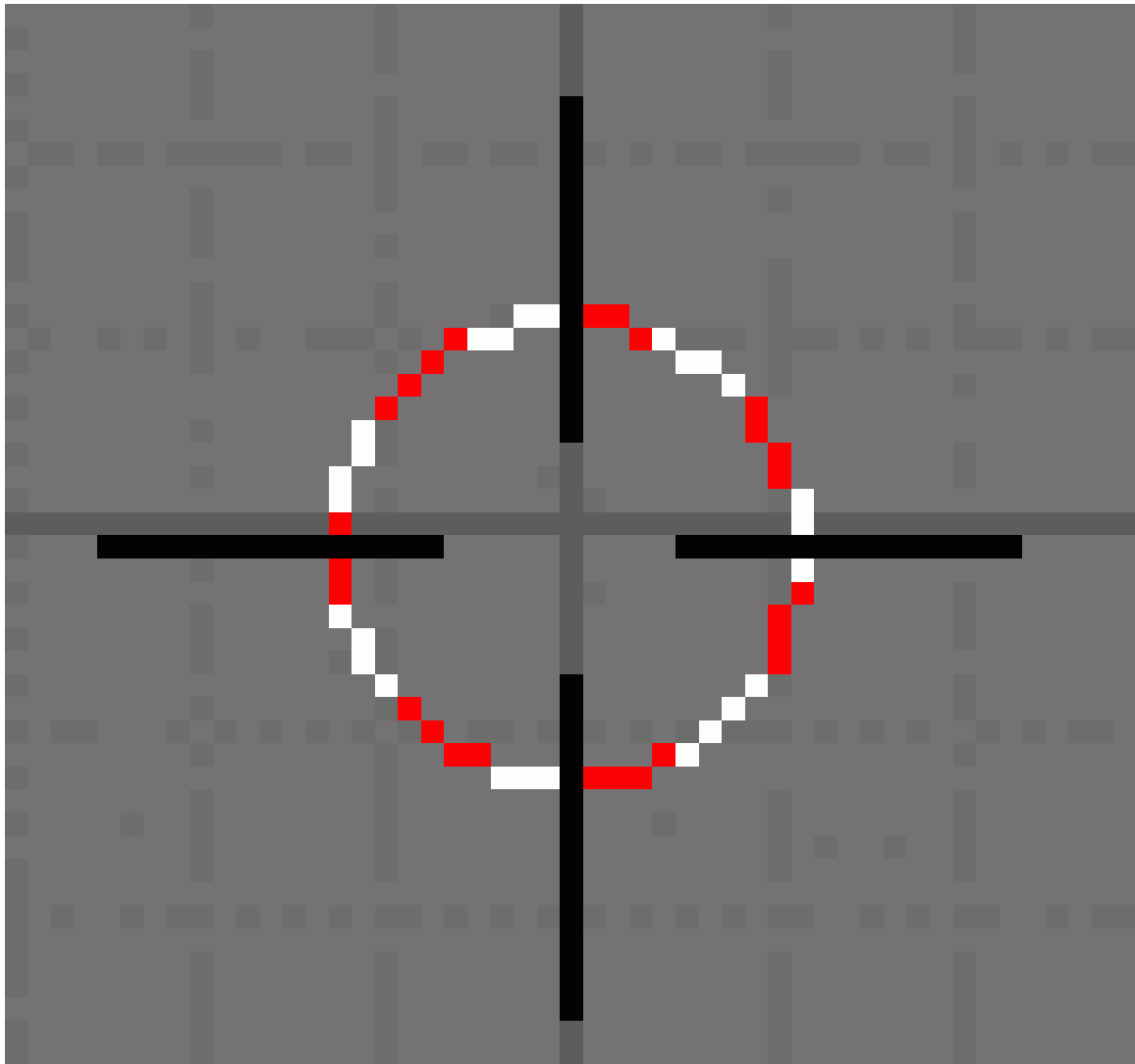


Figure 211

¹ http://en.wikibooks.org/wiki/Blender_3D%3A_Noob_to_Pro%2FBeginning%20Tips

Put the 3D cursor at the scene center by pressing **[SHIFT] + [C]** .

Note: after deleting the cube you must be in Object Mode. If not, switch with **[TAB]** .

46.2. Creating the body

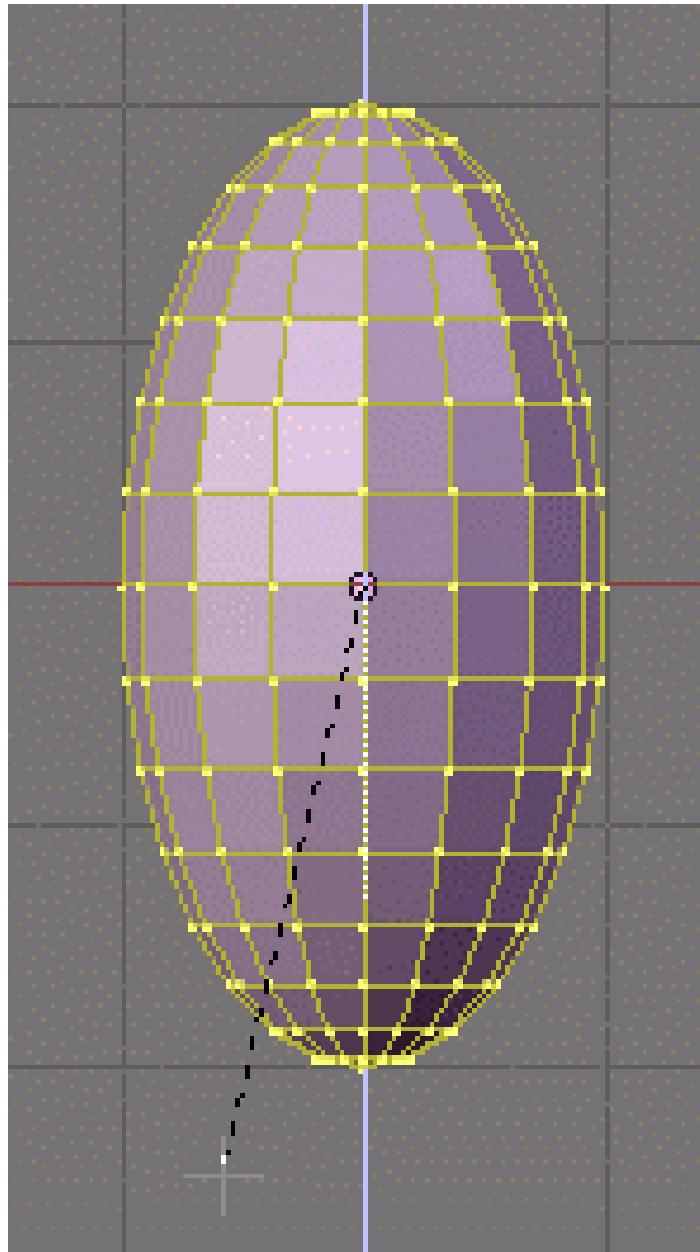


Figure 212 Z-scaled sphere

Noob note: to ensure that you don't become confused, make sure that your viewport is set up in the same direction you see in these pictures. The colored arrows are red, green, and blue and they control the x , y , and z axes, respectively.)

- We start by creating our main body from a sphere. Press **[NUM7]** to switch to top view, and then **[SHIFT] + [A]** → *Mesh* → *UVSphere*, then choose 16 segments and 16 rings.
 - **Note:** You have to manually switch to the Edit Mode after creating the sphere; do this by pressing **TAB** (In older versions of Blender before 2.45, Blender automatically switches to the Edit Mode. This is OK, since we will start editing the mesh now.)
There was a report that in 2.61 it didn't have a pop up unless the **f5** was hit immediately after placing the sphere

We're going to make it look like a penguin body:

- press **[NUM1]** to switch to the front view,
- with all the vertices selected (if not, **[A]**), choose the scale tool (**[S]**),
- restrict scaling to the Z-axis (**[Z]**),
- and move the mouse away from the 3D cursor while holding down the **[CTRL]** key (this snaps the scale values to whole numbers),
 - **Note:** Make sure the mouse cursor is not too far away from the sphere when hitting the **[S]** or else you may not be able to reach a 2.000 scale value. The scaling steps are proportional to the distance from the 3D cursor when calling the scale tool.
- the current scale shows in the lower left corner of the viewport, click when you've reached 2.000 **[LMB]**.
 - **Note:** You can also do this by typing in 2 after starting the scale and restricting movement to the Z-axis

This is our main body!

46.3. Shaping the head

We're going to shape the penguin head from the top of the sphere.

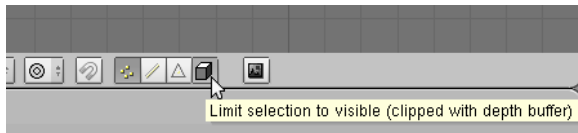


Figure 213

Start by selecting the top-most single vertex as well as the top two smallest circle segments.

Note: selection has been explained in a previous tutorial². Here, the easiest methods are either box selection (**[B]**) in the front view (**[NUM1]**) and Limit selection to visible off, or lasso selection (**[CTRL] + [LMB]**) in the top view (**[NUM7]**) and Limit selection to visible (May also be labeled Occlude background geometry) on. Don't forget to deselect all first (**[A]**)

Noob note: make sure the *Limit selection to visible/Occlude background geometry* button is in the right state each time you select vertices, edges or faces. When it's off, selection affects any item, visible or not.

46.3.1. Building the neck with the 3D transform manipulators

To turn on the 3D transform manipulator, either push down its button

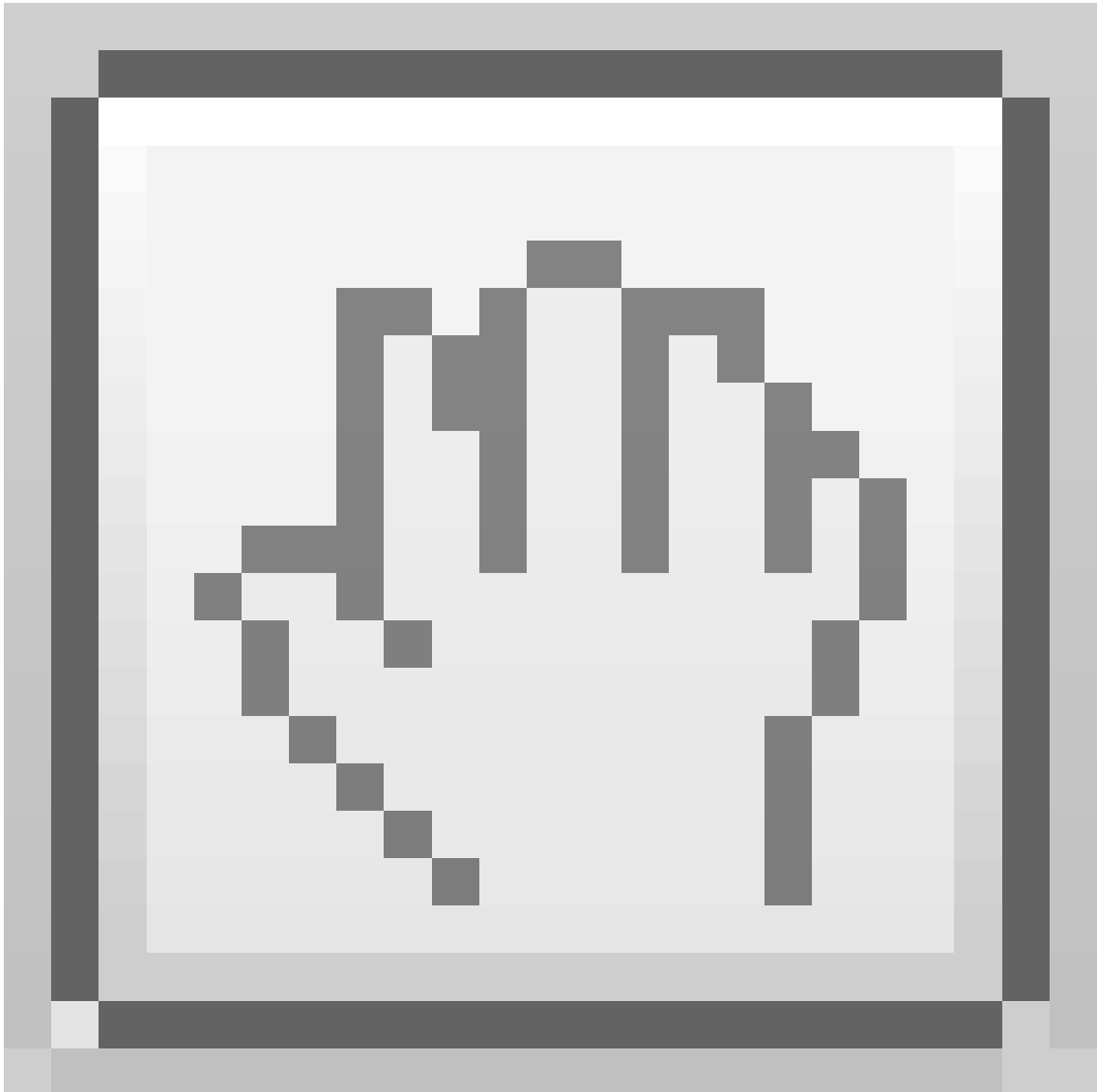


Figure 214

or use **CTRL** + **SPACE** and choose *Enable/Disable*.

Moving the two selected circles up

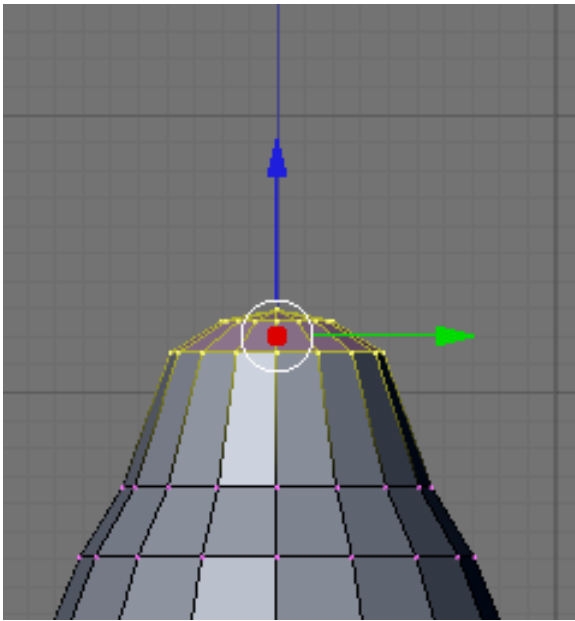


Figure 215

Choose the *Translate* manipulator (red triangle). Go to the front view (`[NUM1]`). Drag the blue arrow while holding the `[CTRL]` key down to move the selected vertices 0.3 units up.

Noob note: you may not be able to snap the extrusion lengths to tenths of units. `[CTRL]` snaps to the grid size by default: if you can only translate by one unit (1.0), zoom in until the grid divides itself into tenths (**SCROLL**). Some Blender versions allow to snap to one tenth of the current step by holding both the `[SHIFT]` and `[CTRL]` keys while moving the mouse.

Noob note: instead of the *Translate* manipulator, you can use the `[G]` and constraint the movement to the Z axis (`[Z]`).

Rotating the neck

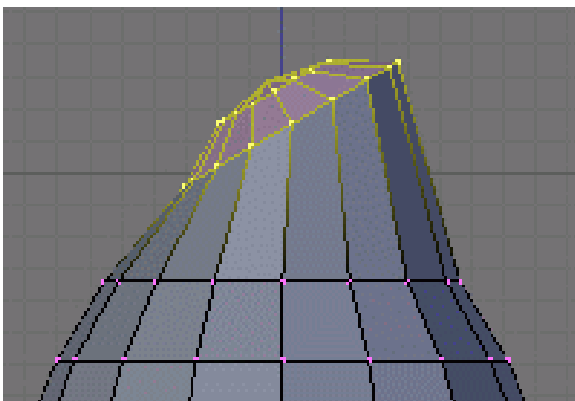


Figure 216

Now switch to the side view (**[NUM3]**) and make sure that the rotation/pivot point is set to "median point" either by selecting it from the third drop down menu right of the "Mesh" menu, or by pressing ";" . Choose the *Rotate* tool (**[R]**). Move the mouse with the **[CTRL]** key down to rotate the selection 30 degrees counter-clockwise. Use **[LMB]** to validate the rotation.

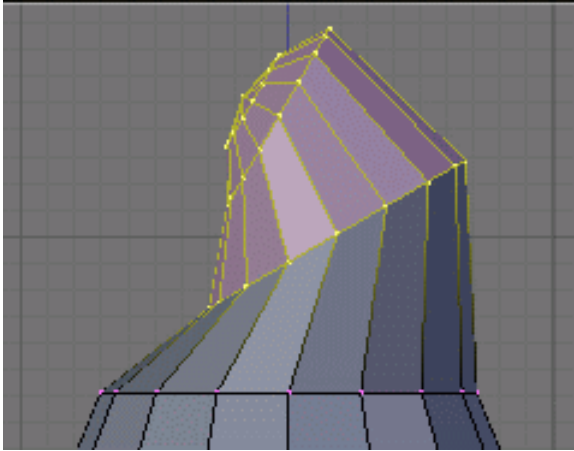


Figure 217

Select an additional ring of vertices by expanding the selection (**[CTRL]** + **[NUM+]** (*Note: NUM+Refers to the addition symbol on the NUM Pad, KEY+will not do.*). You can contract the selection by pressing **[CTRL]** + **[NUM-]** . Move these vertices an additional 0.3 units up, then rotate them as previously 30 degrees counter-clockwise in the side view.

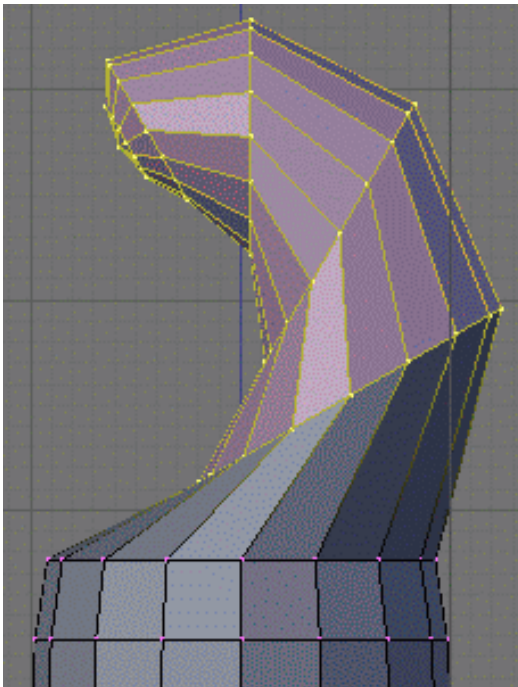


Figure 218

Repeat those steps (selection expansion, translation and rotation) two more times and you'll end up with the body seen in the below.

That doesn't really look like a penguin, yet!

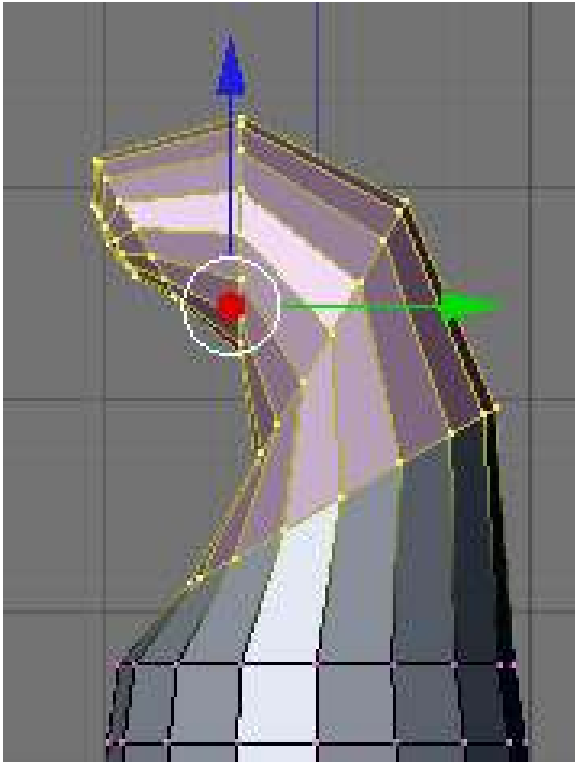


Figure 219

Now move all of the selected vertices to the left 0.4 units by pulling the manipulator's green arrow (and of course holding the **CTRL** key down). This straightens out the neck as seen in the next picture.

Noob note: you can also use the **G** and translate the selection by -0.4 as displayed in the bottom left corner of the viewport. Still do this in the side view (**NUM3**).

Noob note if you had to pull the red arrow, not the green one, then you probably didn't switch to side view, and modeled the neck in front view. If you realize here that you have this done from another view then press **AKEY** twice, then **NUM7**, then **Space** -> **Transform** -> **Rotate** and type **90** or **270**

Creating the beak

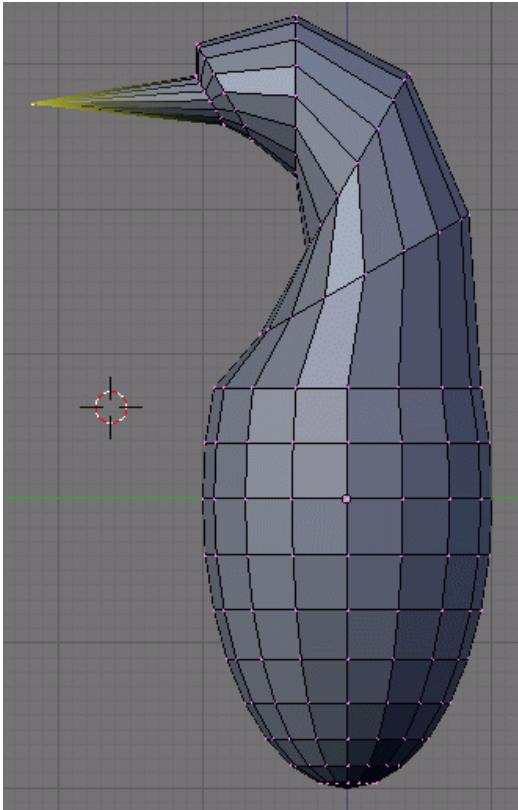


Figure 220

Switch to the front view (`[NUM1]`), and select the frontmost vertex (the one that originally was the top vertex of the sphere) with the `[RMB]` . Then switch to the side view (`[NUM3]`) and translate this vertex to the left by 1.2 units using the manipulator's green arrow or the translate tool.

Note: some Blender versions allow moving the vertices from the keyboard with the following sequence: `[G]` , `[Y]` , `-1.2` , `[ENTER]` .

The main body of the penguin is now finished. The next step is to create some flappers for our poor little guy.

46.4. Extruding the wings

We are going to create the wings by extruding faces on each side of the penguin.

Noob Note: You can rotate the whole object by pressing `A` to select all of its vertex and then rotate it pressing `R` or using the Rotate Manipulator until the axis on your screen matches the axis on the example image. That way, also, you can practice a little more

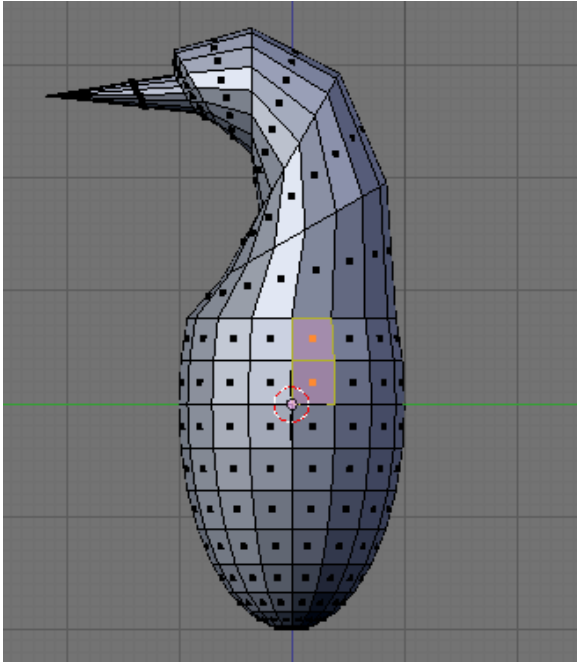


Figure 221

Choose the side view (**NUM3**) and switch to the *Face select mode* (**CTRL** + **TAB** → *Faces*, or click on the triangle in the toolbar).

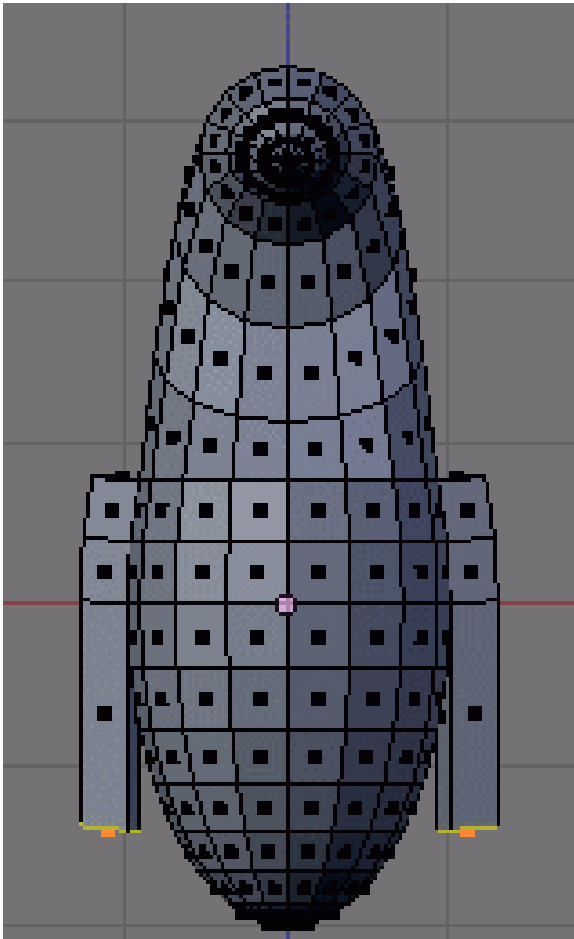


Figure 222

Now, select two faces that will make up the penguin's shoulder as shown on the right (either select the first one with **[RMB]** and the second one holding **[SHIFT]**, or use box selection (**[B]**) to select them both in a single operation).

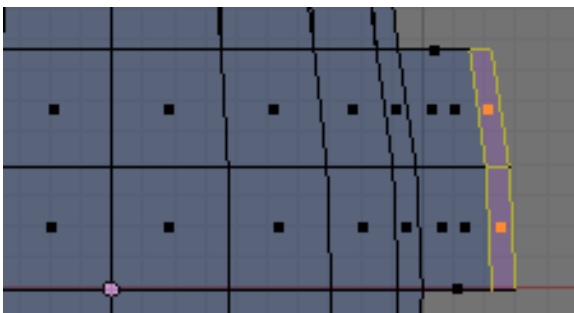


Figure 223 Wing extrusion

Then switch to front view (**[NUM1]**) and extrude the selection:

- choose **[E]** → *Region*,
- constrain to the X axis (**[X]**),

- hold **CTRL** to snap,
- and move the mouse to extrude the shoulder by 0.2 to 0.3 units.

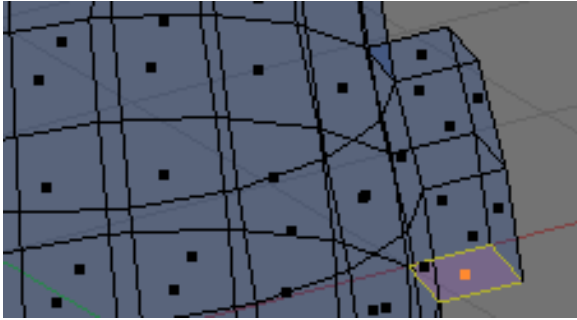


Figure 224 Bottom face selection

We'll now extrude the bottom face of this new extrusion. Rotate the view to show it with:

- a **MMB** drag,
- or several presses on **NUM2**,
- or **CTRL** + **NUM7** (bottom view).

Press the **A** to deselect all, and select the bottom face (**RMB**), switch to front view (**NUM1**), extrude by 1.4 units down (**E** and **CTRL**).

Now do the same on the penguin's other side: use **CTRL** + **NUM3** to view the left side (you can also rotate with **MMB**, or press **NUM4** several times).

46.5. Smoothing the wings

We're going to smooth out the shoulders and improve the wings. Though this can be done in many ways, we'll only use the merge tool.

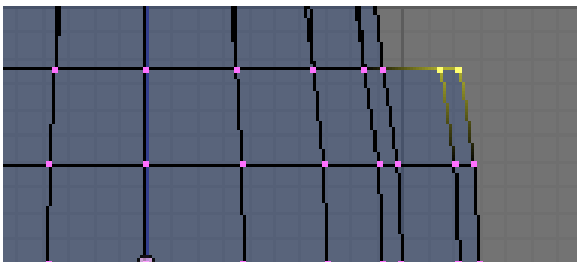


Figure 225

Rotate your penguin so that you can see one shoulder from above. Then switch to *Vertex select mode* (**CTRL** + **TAB** → *Vertices*). Press **A** to deselect all, then select the two shoulder vertices with **RMB** and **SHIFT**.

Press **ALT** + **M**, choose *At Center* from the popup in order to merge the two vertices at their center. Finally dismiss the message saying *Removed 1 Vertices*.

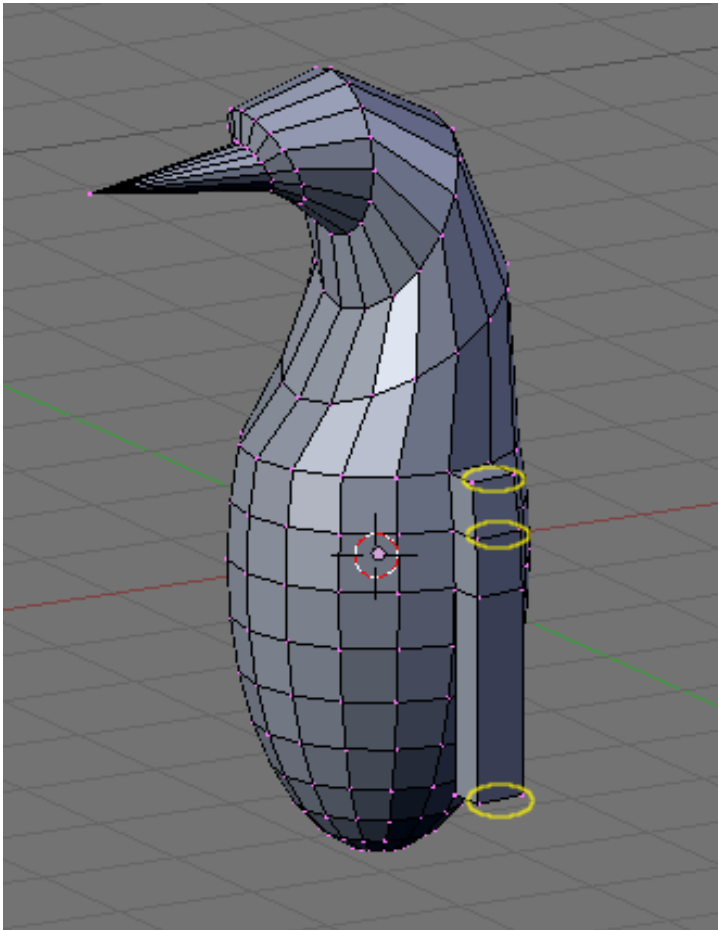


Figure 226 Edges to smooth

Repeat the steps with the two other vertex pairs shown on the right picture, and smooth the other wing. I'm leaving the middle segments for now, else the wing tips will be too pointy.

***Note:**if you have troubles merging vertices, it comes from vertex duplicates in your mesh. You probably chose Individual Faces instead of Region when extruding the wings, which creates duplicate vertices and neighbouring faces. To clean up your model: select all vertices (**A**) and choose **W** → Remove Doubles.*

You must do them one by one!

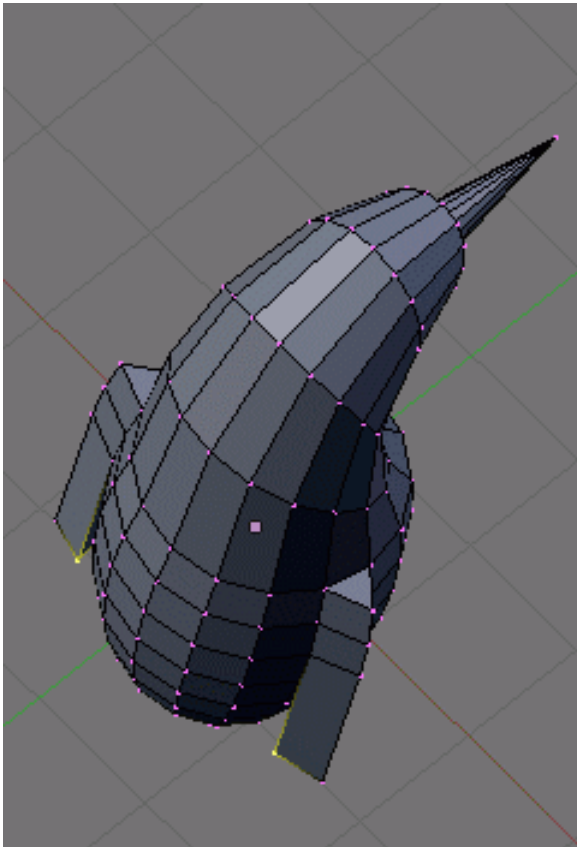


Figure 227

Finish off the wings by selecting the two backmost vertices of the wings, and moving them up using the blue arrow by 0.1 unit.

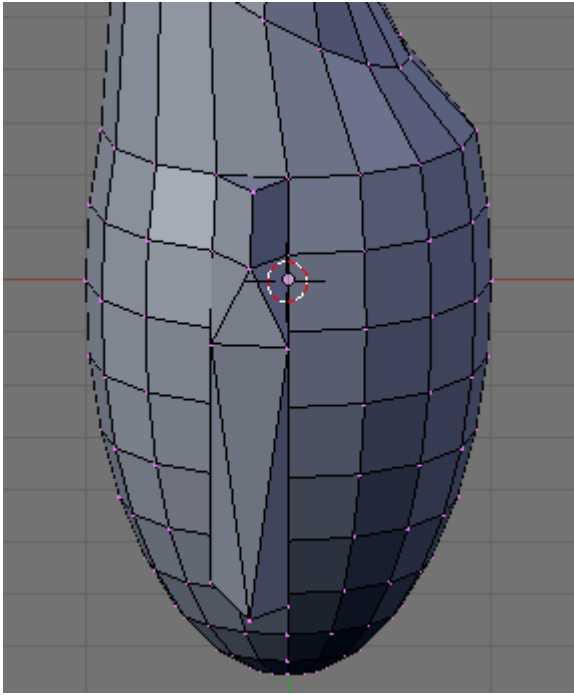


Figure 228 Wings complete

You should now have something like this:

46.6. Cutting the underside

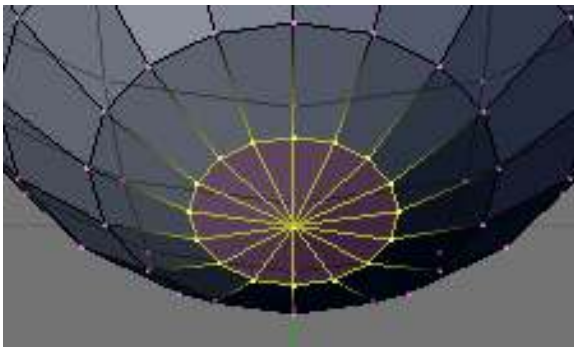


Figure 229

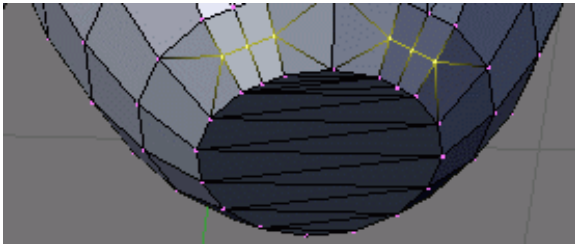


Figure 230

We're going to cut the penguin's lower end, for it to stand up! Select the bottom vertex (bottom vertex and the first ring above it) as shown in the picture. There are many ways, this is left as an exercise.

Once they're selected, delete them (**X** → *Vertices*). Now our penguin is hollow: select all the vertices around the hole, and fill it using **ALT** + **F**.

Noob note: For Blender 2.56 **SHIFT** + **F** goes into Camera Fly mode, **ALT** + **F** however did the filling.

Noob note: To quickly select all the vertices around the hole, you can enter edge mode **CTRL** + **TAB** → "Edges", and then select one edge that goes around the hole. Now press **CTRL+E** → "Edge Loop Select", which should select all edges around the hole. Now go back to Vertex select mode and continue with **SHIFT** + **F**.

Noob note: For Blender 2.56 In vertex select mode and occlude on hold down **ALT** and select one of the vertices all the vertices in the ring will be selected.

46.7. Adding the feet

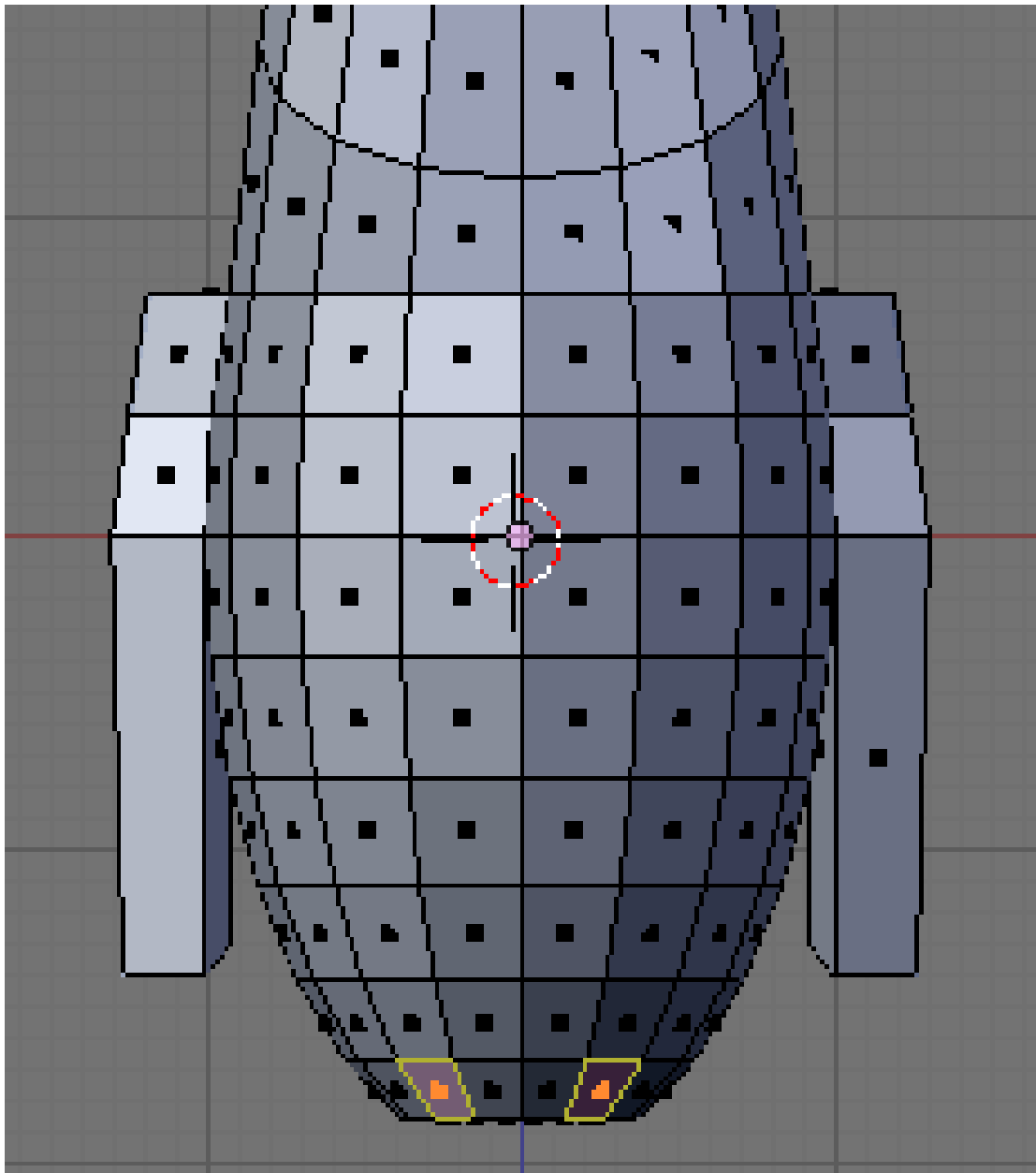


Figure 231

The next step is to provide the little guy with feet. To do this, we're going to extrude two of the front faces:

- choose the front view (),
- switch to *Face select mode*,
- turn on *Limit selection to visible* (labeled *Occlude background geometry* in later versions of Blender)
- and select the face to the left and right of the middle two faces of the penguin.

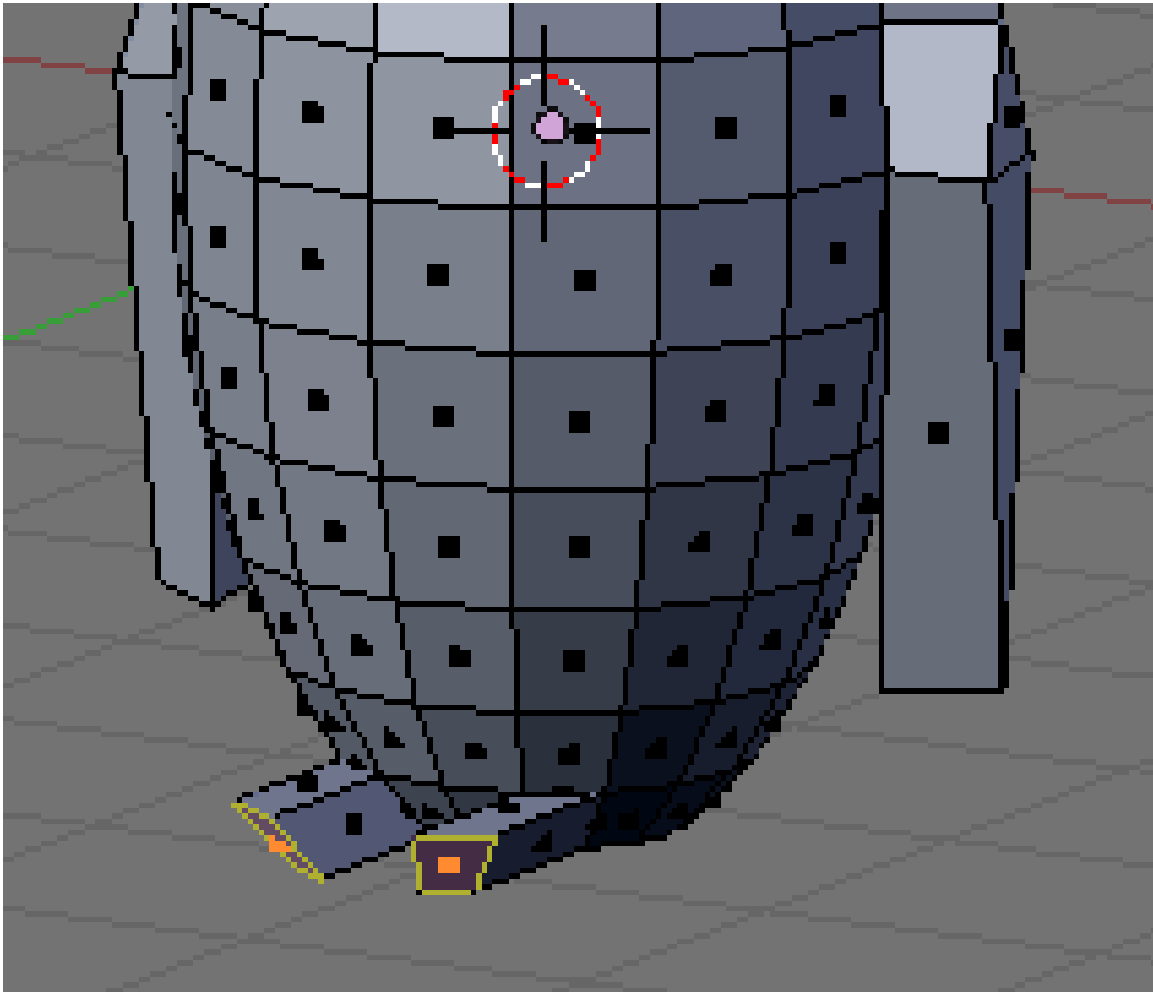


Figure 232

Then switch to the side view (**NUM3**) and extrude the selection by -0.6 units (**E** → *Region*, restrict to the Y axis: **Y**).



Figure 233

Keep the selection and look for the *Mesh Tools* in the *Buttons* panel. If you can't see it, press **F9** to switch to this panel. Then click on the *Subdivide* button (at the top, between *Short* and *Innervert*). Or press **W** and choose *Subdivide*

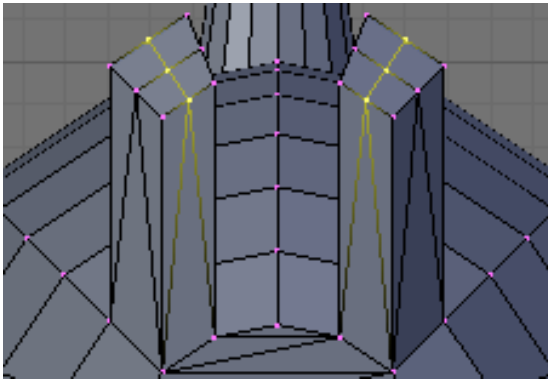


Figure 234

Switch to "Vertex Select Mode", Now select the three middle vertices (or two edges) vertically at the tip of each foot, and drag them along the Y axis by 0.3 units towards the penguin

Note: if something goes wrong here, you may need to remove double first. As always, to move the vertices, either use the manipulator or **G** and **Y** sequence.

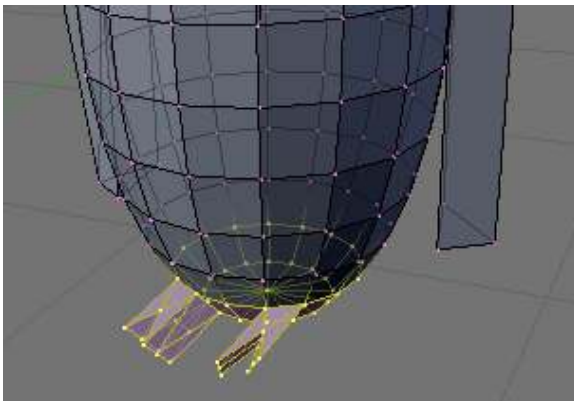


Figure 235

You should end up with what's shown in the right picture (minus the selection).

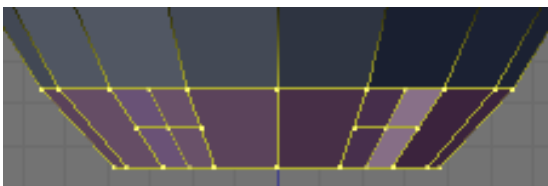


Figure 236

The feet look too thick, let's flatten them a bit. Switch to the front view (**NUM1**) and select the two bottom vertex rows (*Limit selection to visible* off, use either the lasso or box selection).

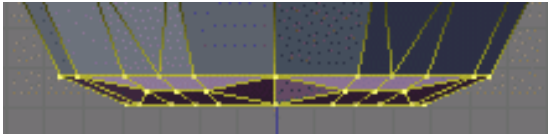


Figure 237

Then choose the scale tool (**S**), limit its action to the Z axis (**Z**) and scale down by a factor of 0.4.

The feet still look rather peculiar, so please go ahead and move the vertices around on your own as you like.

Reminder: you can use the **G** and restrict movements to the X or Y axis using the **X** or **Y**. Try not to move vertices along the Z axis to keep the penguin's bottom flat.

46.8. Extruding a tail

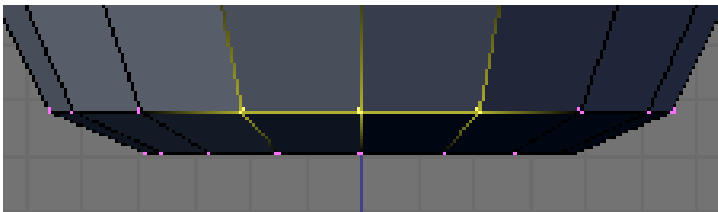


Figure 238

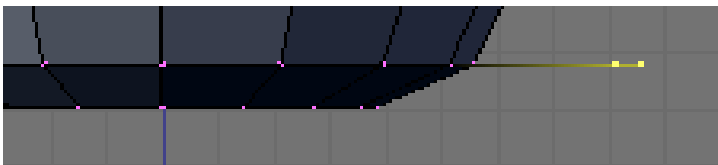


Figure 239

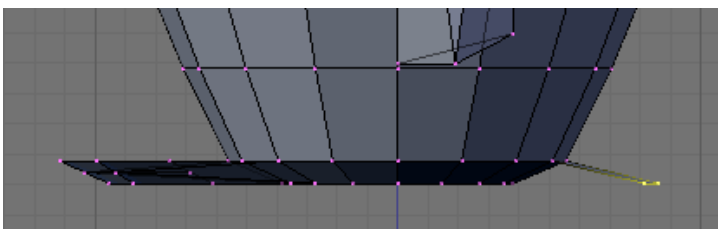








Figure 240



To complete the penguin, we have to add a tail (the end of the tuxedo):

- go to the back view (**CTRL** + **NUM1**),
- make sure you're still in *Vertex select mode*,

- and that *Limit selection to visible/Occlude background geometry* is on.

Select the three middle vertices in the second row up from the bottom. Then, switch to the side view () and extrude the edges 0.3 units away from the penguin and 0.08 units down ( → *Edges*), so that the end of the tail is at the same level as the bottom of the penguin.

Noob Note: Or you can extrude the edges 0.3 units away from the penguin, then   -0.08,  or  .

Press  +  to save your work!

46.9. Subsurfing

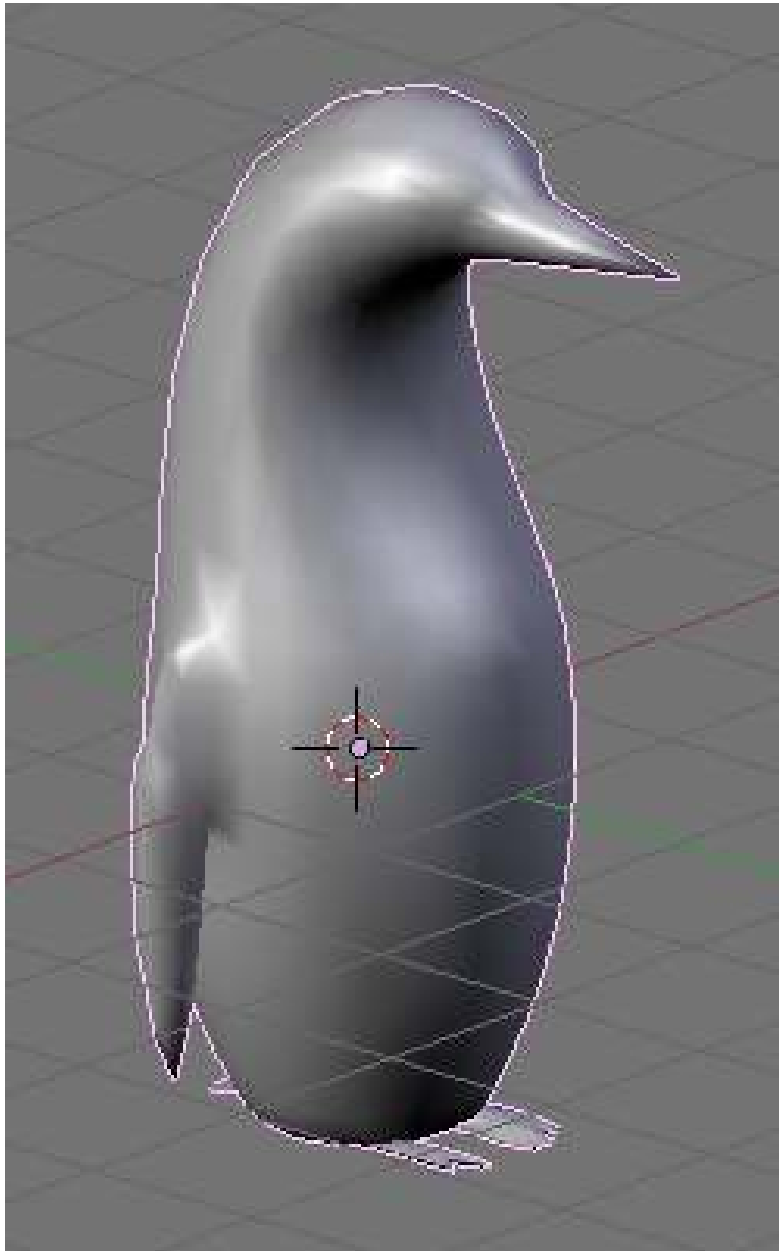


Figure 241 Now that's what I call a penguin. But then again, there are no penguins where I live, so I might be wrong.

Go to *Object Mode* (`[TAB]`), and make sure the penguin is selected. Then check for the *Modifiers* toolkit in the *Buttons* panel. Press *Add Modifier* → *Subsurf* (or Press `[SHIFT] + [O]`).

Look at the penguin now, he's much smoother. You can alter the levels of the subsurfing if you like, but I'll settle for level one. Under the *Links and Materials* toolkit, you can press the *Set Smooth* button as well, which makes the penguin really slick.

Note: you may see some weird effects at the bottom and the tail after subsurfing the penguin. If so, there is an issue with normals: they have to be all pointing outwards. This can be achieved by selecting all vertices in Edit Mode and recalculating the normals outside (`CTRL` + `N`). Click on the message to confirm. Note that `CTRL` + `SHIFT` + `N` will turn the normals inwards and that `W` → Flip Normals flips them.

Question: The finished penguin looks fine in Object Mode, but when I render it, it looks odd. Quite patchy.

Answer: Make sure you adjust the "Render Levels" parameter (directly under "Levels") to be greater-or-equal than "Levels".

46.10. Extra

The penguin can be colored or textured, but that will be part of later tutorials!



Figure 242 This is what the penguin (sans tail) looks like, textured and ready. Orbisonitrum^a

^a <http://en.wikibooks.org/wiki/User%3AOrbisonitrum>



Figure 243 The eyes are there, just not easily visible in the thumb. At the top of the white part, two faces on the chest were subdivided to give the white more of a curve at the top. The faces were selected that were going to be white, and the I used separate (**P**) to make them a different mesh. I used a white material for the chest, black for the body, and grey uvspheres for the eyes. Apparently, an easier way to colour the chest can be found at *Multiple Materials^a please feel free to replace this with your own image of the penguin you made, with comments on how you put your own style into it --Most recent is Bnty^b 11:35 PST, 19 May, 2006*

^a <http://www.blender.org/documentation/htmlI/x4448.html>

^b <http://en.wikibooks.org/wiki/User%3ABnty>



Figure 244 A pretty basic picture of a penguin. I subdivided the stomach and eyes, but then I also added some eyeballs by making a UVSphere, cutting the top of it off, and then placing it inside of my penguin's head. All the colors have specular colors, giving the penguin a slight blue glow under the black.

47. Die Easy 2

The previous *Die Easy* tutorial needed so many changes to adapt to Blender 2.44 that it was easier to start from scratch. This *Die Easy 2* tutorial is heavily based on *Die Easy* and *Die Another Way*, and adds improvements.

In the following tutorial, you will use:

- polygon mesh
- face loop cutting
- subdivision surfaces
- subdivision creases
- bevel
- set smooth
- multiple materials
- extrusion
- merge vertices

47.1. Start with a beveled cube

Start with the default scene¹: Blender should be in *Object Mode* and in *Top* view, with the default cube selected.

- Go to *Edit Mode* (**TAB**). All the vertices should still be selected. (If not, press **A** to select all).
- Let's bevel the cube. The *bevel* option is in the **modifiers context**. Click on the modifiers icon (the little blue wrench on the side panel) and from the *add modifiers* choose **Bevel**.
 - **Noob note** :- for anyone using v2.56 and below, Press **W** , select *Bevel* from the menu and click *OK*
- Set the bevel to 0.1 by typing *.1* and pressing **ENTER** .
- Finally, **Apply** the modifier to your cube before proceeding.
- For Blender versions earlier than 2.48 : in the *Recursion: 1 OK* dialog, Press **ENTER** (it will revert to the black on grey) and click on the *OK* button. (For Blender version 2.48 and above, skip this step)

Note : The recursion levels go from 1 to 4. At 1 it seems to be crude but the cube will be subdivided later, so a higher recursion level is not required.

¹ http://en.wikibooks.org/wiki/Blender_3D%3A_Noob_to_Pro%2FBeginning%20Tips

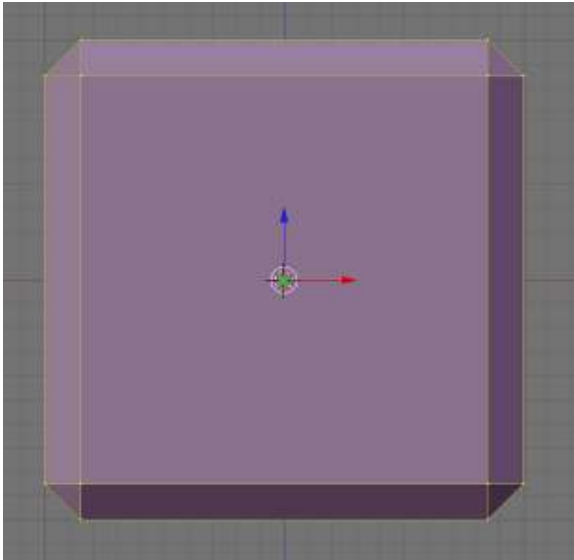


Figure 245

47.2. Divide the faces

A typical die face is made of a grid with 9 possible spot positions. The gaps between the spots usually measure half a spot (a spot radius). Using the spot radius as a unit, a die face will be conveniently divided into 10 equal parts through the length (that is, four gaps and three spots each two units long).

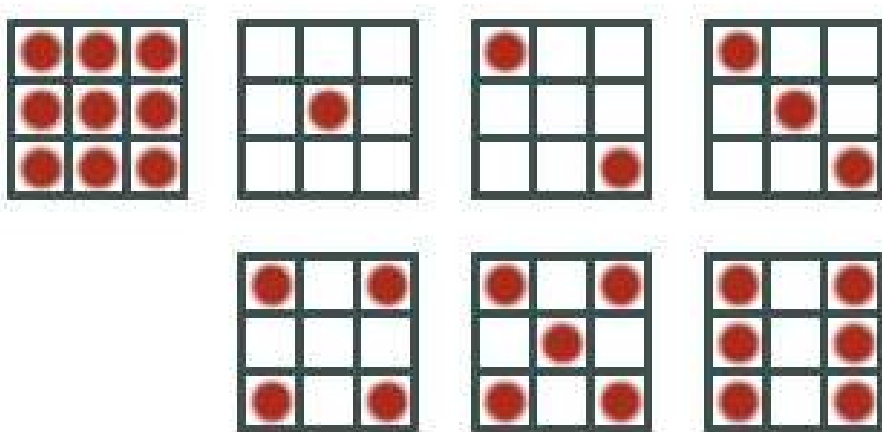


Figure 246

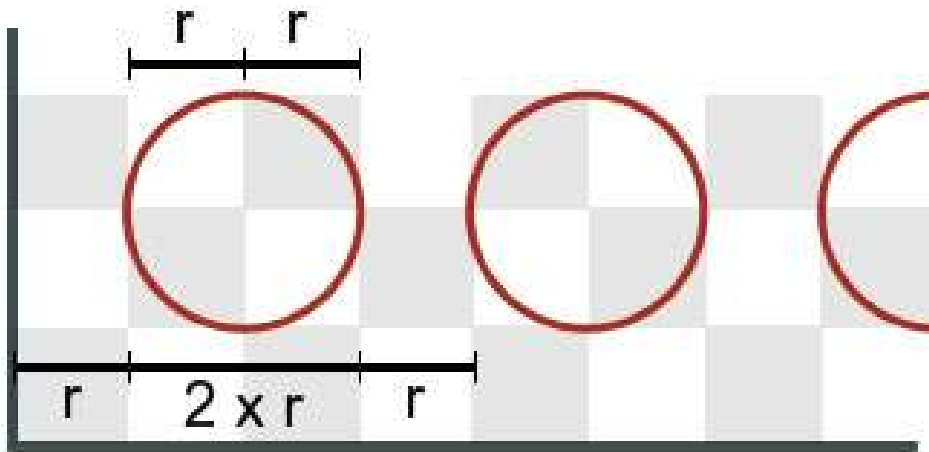


Figure 247

47.2.1. First division

- Make sure you're still in *Edit Mode* (else hit **[TAB]**), that you're in *Front view* (**[NUM1]**), and that the whole cube is still selected.
- To divide the faces of the cube into 10 vertical strips :

For v2.57 and above :

- Just use the **loop cut and slide** option by hitting **[CTRL] + [R]** in the tool shelf to the left and scroll the mouse wheel to scale up the number of cuts to 9.

For other versions of Blender :

- Press the **[K]** and choose *Knife (Multicut)* from the popup menu.
 - A new popup asks: *Number of Cuts: 2*. Change the value to 9:
 - click on the *of* word and type 9,
 - or drag from the input field (**[LMB]**) until you reach 9,
 - or use the right arrow.
 - Then click on *OK*.

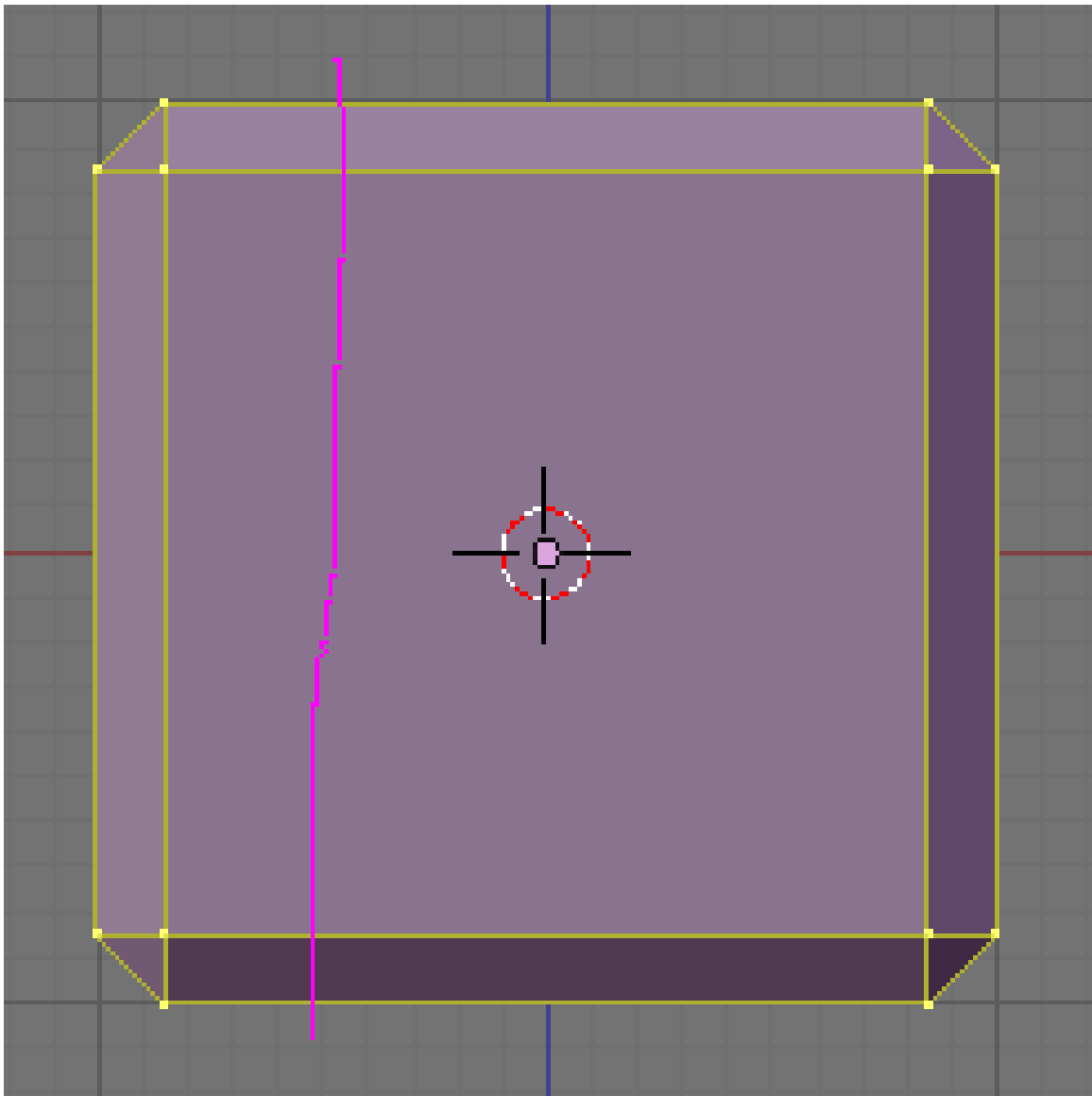


Figure 248

- To actually use the *Multicut* tool, draw one vertical cut through the cube: the knife icon traces a pink line. It doesn't have to be exactly vertical nor does it have to go through the cube center.

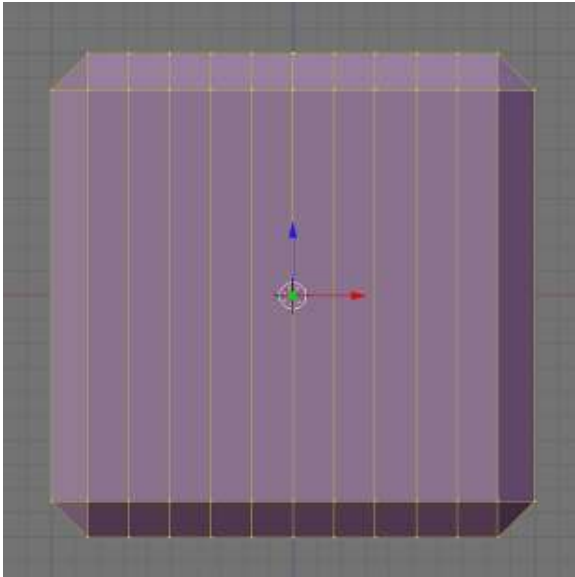


Figure 249

- Finally press **[ENTER]** and all the faces of the cube crossing the knife will be divided into 10 vertical strips.

Note: You can have a look at all the other faces by dragging with the **[MMB]**.

Noob Note: When you are cutting through the cube, make sure to cut the entire cube, including the beveled part of the cube. Otherwise your cube won't divide properly.

47.2.2. Remove extra cuts

To get each spot position to be a single face rather than four contiguous faces, the second, fifth and eighth cuts must be removed.

- Press the **[A]** to deselect all the vertices, switch to *Edge select mode* and make sure *Limit selection to visible* is off (in Blender 2.46 and above this button is called *Occlude background geometry*).

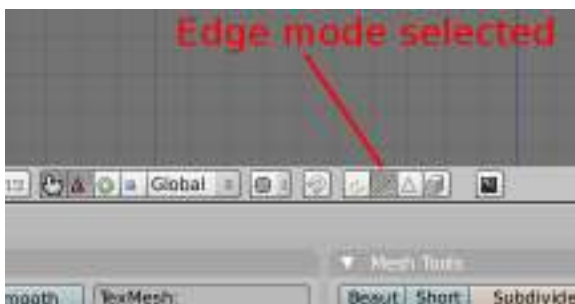


Figure 250



Figure 251 The button is disabled (off)

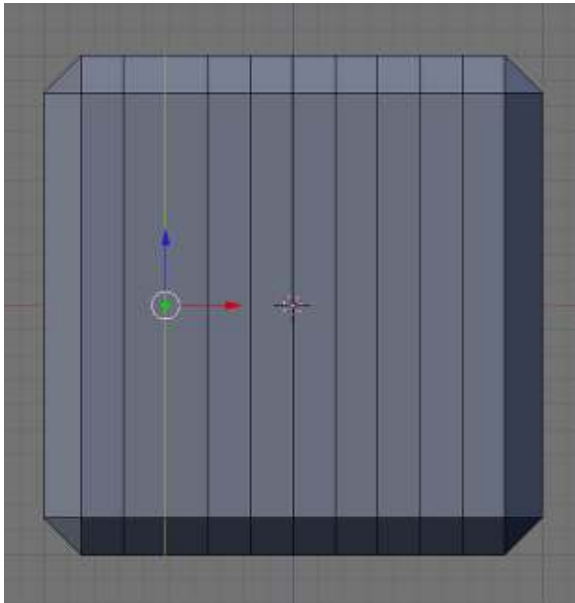


Figure 252

- Select an edge from the second cut, and choose *Select* → *Edge Loop* from the viewport menu. This automatically selects all the edges around the cube (in a loop).

Note: You can alternatively press the **[ALT]** key while selecting an edge from the second cut to select all the edges belonging to this cut around the cube.

For Linux users: If pressing the **[ALT]** key while selecting an edge from the chosen cut brings up a Linux menu, try hitting **[SHIFT] + [ALT]** instead.

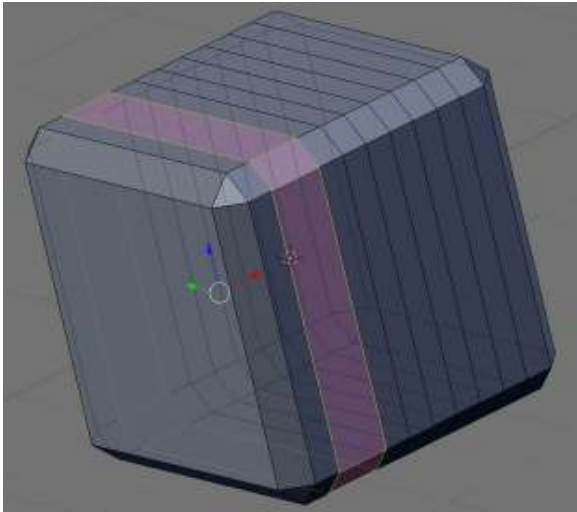




Figure 253

- Press the  and select *Edge Loop* from the popup menu. The cube will look like this:
- Deselect all () and repeat the same actions to remove the fifth and eighth cuts:

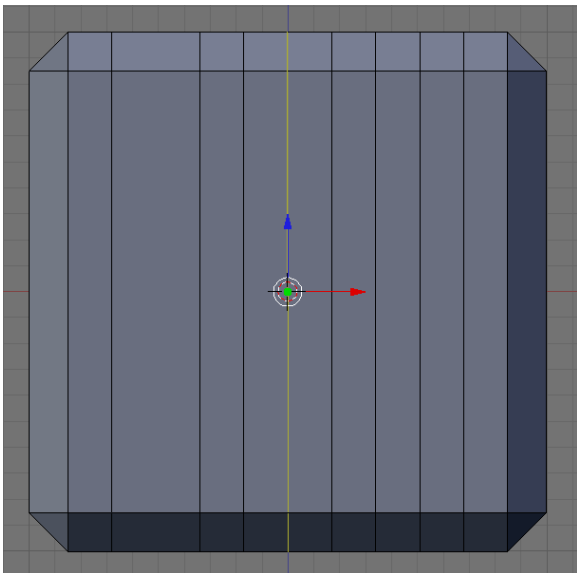


Figure 254

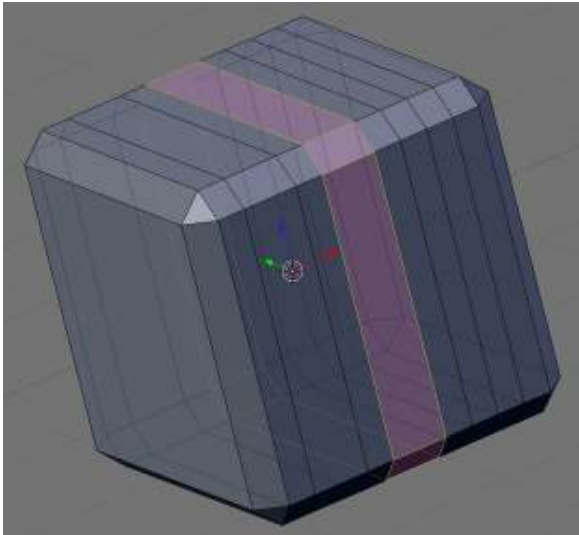


Figure 255

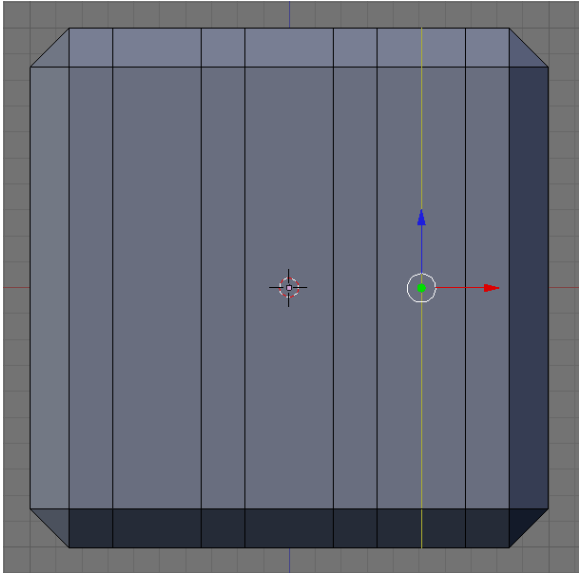


Figure 256

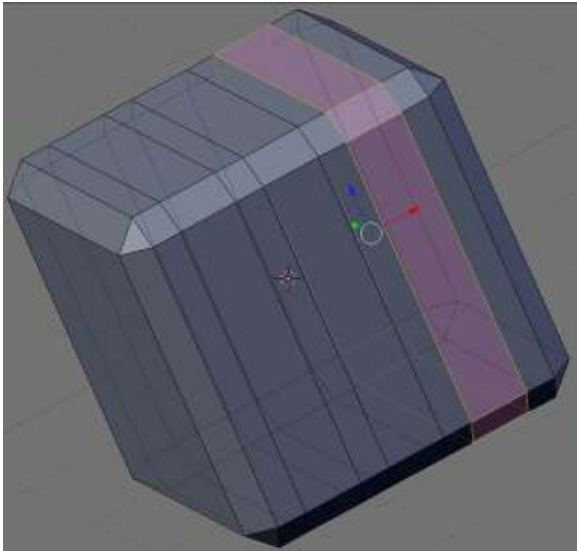


Figure 257

47.2.3. Other divisions

For the horizontal cuts use the same steps as above, but horizontally!

Noob Note: Make sure the whole cube is selected when cutting.

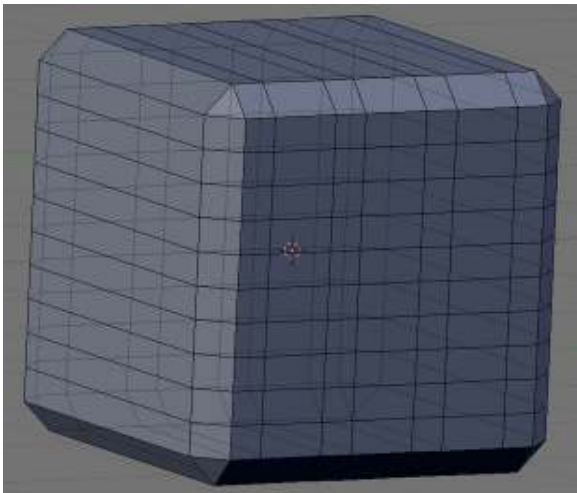


Figure 258

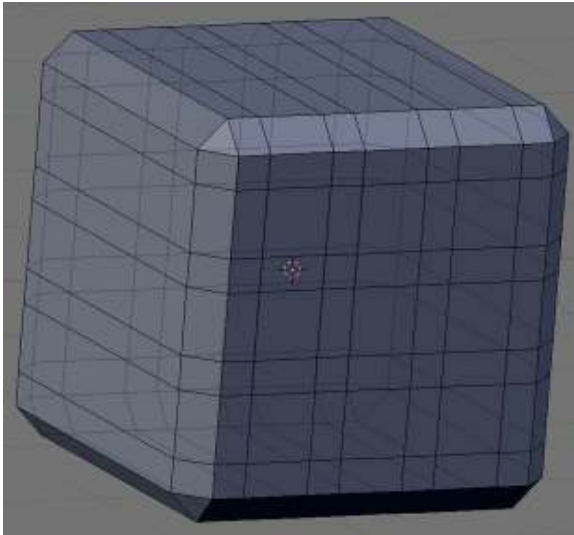


Figure 259

- Now, if you rotate the cube around, you will see that the front and back faces (**CTRL** + **NUM1** to see the back face) are divided vertically and horizontally. But the four other ones were only divided once.
- To complete the divisions, go to the *Side* view (**NUM3**), select the whole cube (**A**). Make the vertical cuts and remove the second, fifth and eighth cuts again:

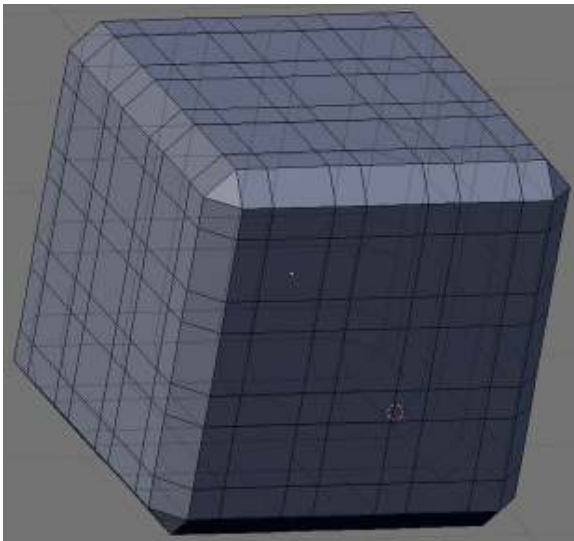


Figure 260

47.3. Colouring the Cube

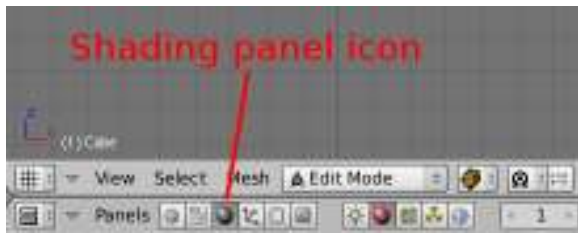


Figure 261

You need to be in object mode for this step. Select the *Shading* group from the header in the *Button Window* (**F5**).

Change the name of the material (which for example defines colours) currently linked to the cube: in the *Links and Pipeline* panel, under the title "Link to Object", click on "Material" in the "MA:Material" box. "Material" will now be highlighted: delete it and type "DieWhite" instead, then press **ENTER**.



Figure 262

Now choose a color: go to the next panel (the one with two tabs, *Material* and *Ramps*), and make sure the *Material* tab is selected.

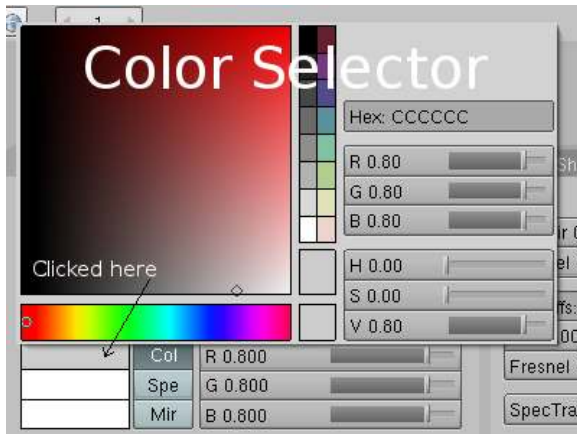


Figure 263

In the second group of buttons, click on the colour bar which is on the left of the *Col* button: it opens the colour selector.

Select the colour you want in the rainbow bar (bottom left of the popup), move the tiny circle in the colour window above it to adjust the shade. When done, press **[ENTER]**.

Note: you can choose any colour. In this tutorial we selected the white colour by clicking on the corresponding box available from the sixteen predefined ones (the two vertical rows in the middle).

47.4. Adding the spots

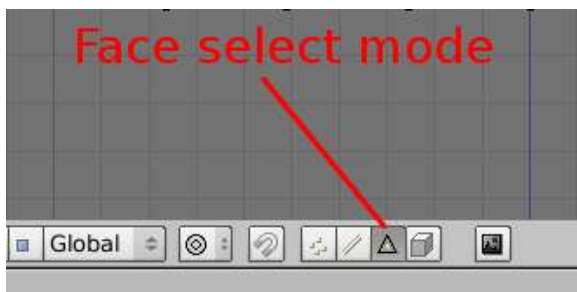


Figure 264

Go to the *Front* view (**[NUM1]**) and set *Face select mode*. Also make sure that *Limit selection to visible* is enabled (in Blender 2.46 and above this button is called *Occlude background geometry*).

We will create two different kinds of spots.

47.4.1. First method

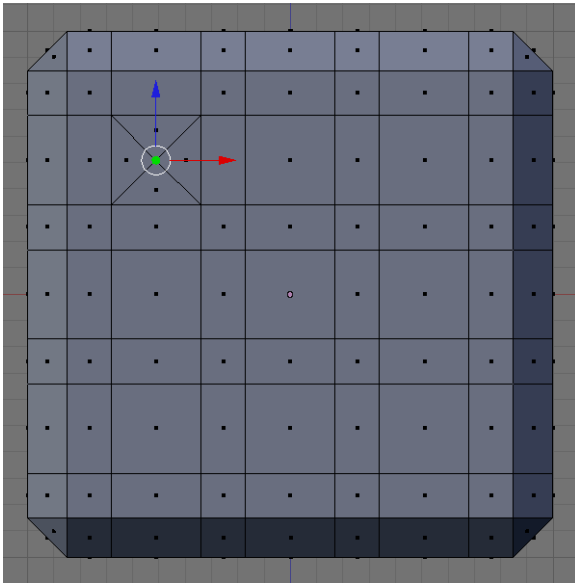


Figure 265

This one's from the *Die Another Way* tutorial.

We'll make the number five on the current face. Deselect all (**A**) and select a first spot (the one in the top left corner).

Extrude the face by hitting **E** and then **ESC** (do not click after hitting **E**).

Collapse this face by using **ALT** + **M** and choosing *At Center* from the popup menu. This merges the four corners of the extruded face into its centre. Click on the message saying *Removed 3 Vertices*.

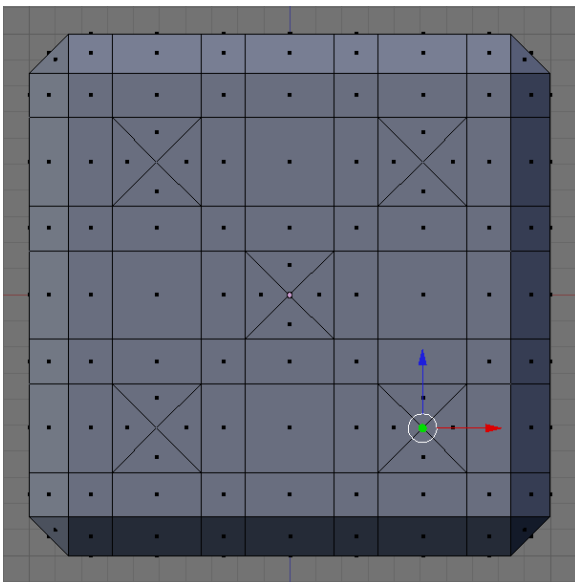


Figure 266

Repeat the same keystroke sequence on the four other spots: **E**, **ESC**, **ALT** + **M**, **ENTER**, **ENTER**.

Now choose the *Vertex select mode* and select five vertices, one at each spot center (**RMB** for the first one, **SHIFT** + **RMB** for the subsequent ones). Then push the selection inwards using the **G** and constraining the move to the Y axis (**Y**). Move by 0.17 units to get the same die as shown in the picture below.

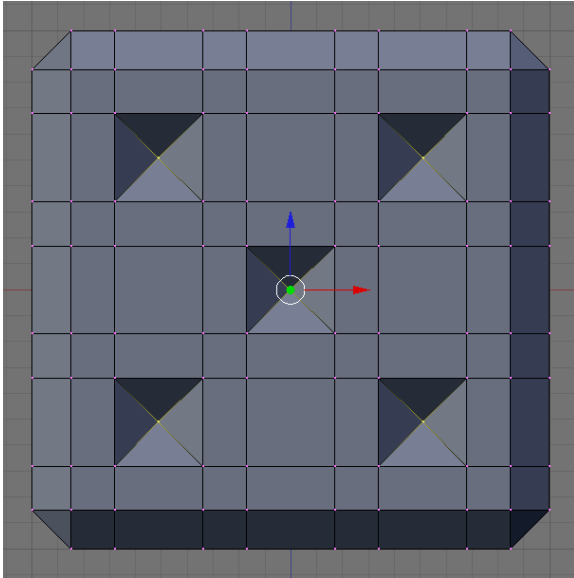


Figure 267

47.4.2. Second method

This method was suggested by Natume in the *Die Easy* tutorial and modified by another user.

Deselect all (**A**) and select the faces you want to create spots on (**RMB** for the first one, **SHIFT** + **RMB** for subsequent ones).

Press the **E** to extrude them, if you selected more than one face a popup menu will give you two options: press **ENTER** to select *Region* (the choice doesn't matter), and then press **ESC**. Do not click after hitting **E**.

Divide the selected faces into triangles with **CTRL** + **T** (or **SPACE** → *Edit* → *Faces* → *Convert to Triangles*).

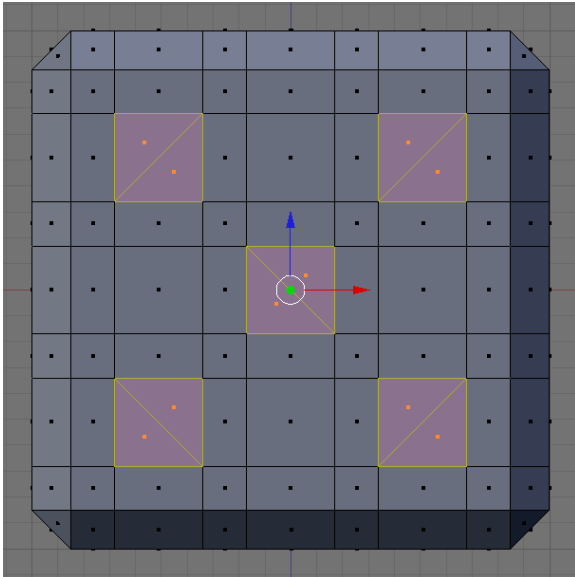


Figure 268

Deselect all (**A**) and change to *Edge select mode*. Select all the diagonal edges resulting from the above divisions, and subdivide them: **W** (or **SPACE** → *Edit* → *Edges* → *Subdivide*).

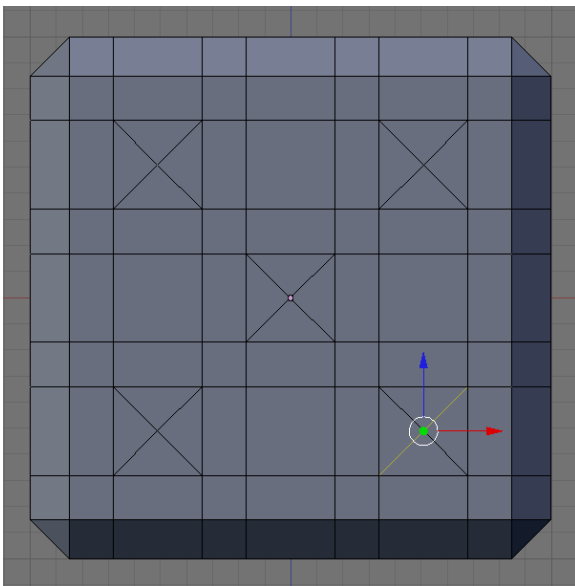


Figure 269

Now deselect all (**A**), go into *Vertex select mode* and select all five vertices of each spot face (for example, use circle select with **B**, **B**, and adjust the size of the circle with **SCROLL**, then click **LMB** to select; click **RMB** when the selection is complete).

Push the faces inward by 0.17 units (**G**, **Y**).

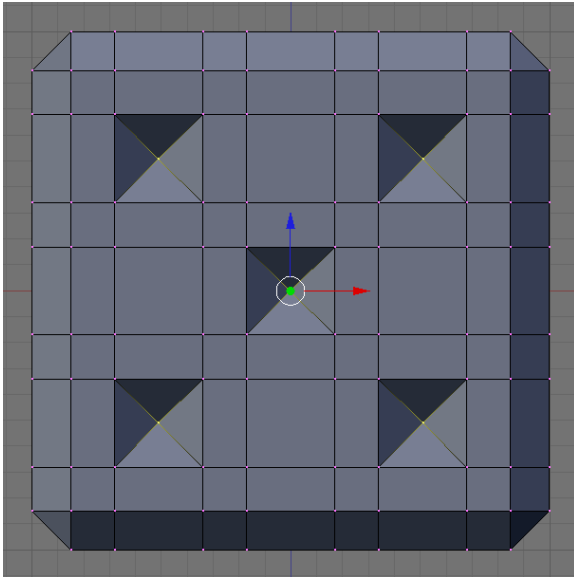


Figure 270

47.4.3. Third method

In the prior two methods you create indentations in the faces that are triangle (mountain peak) shaped, if you would prefer them to be square shaped (or flat) then simply select the faces (using face select mode) that will be used as spots and extrude them inward by the prescribed number of units (either .17 or -.17 depending on the side).

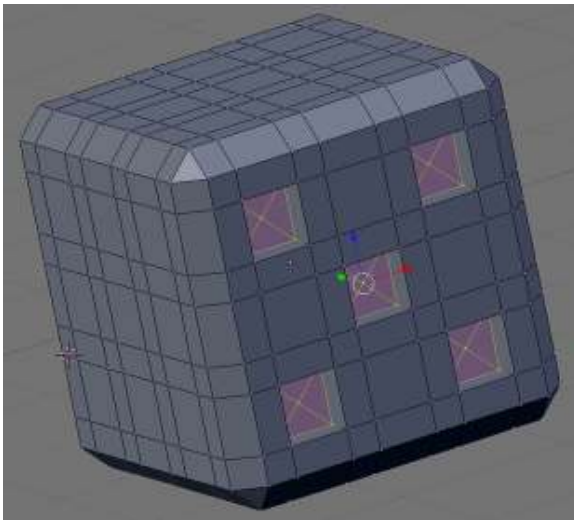


Figure 271

47.4.4. The other die faces

Now repeat the method to the other faces. Remember that adding two opposite face values (of the die) yields 7.

If you started with the *Front* view () and set 5 spots, the *Back* view (+) will have 2 spots (the extrusion length will be -0.17 instead of 0.17).

If you then go to Side view () — the righthand side — and put 6 spots there (extrusion -0.17 along the X axis,), the lefthand side (+) will have 1 (extrusion 0.17).

If you put 4 on the top (Top view , extrusion -0.17 along the Z axis,) then the bottom (+) will have 3 (extrusion 0.17).

When all the spots have been created, the cube should look like this — the top lefthand corner is nearest, using the first method (named *Alt-M*)

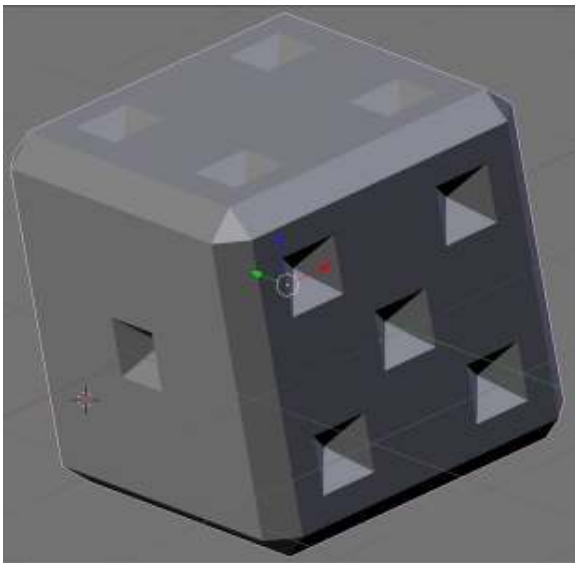


Figure 272

using the third method (named *Ctrl-T*)

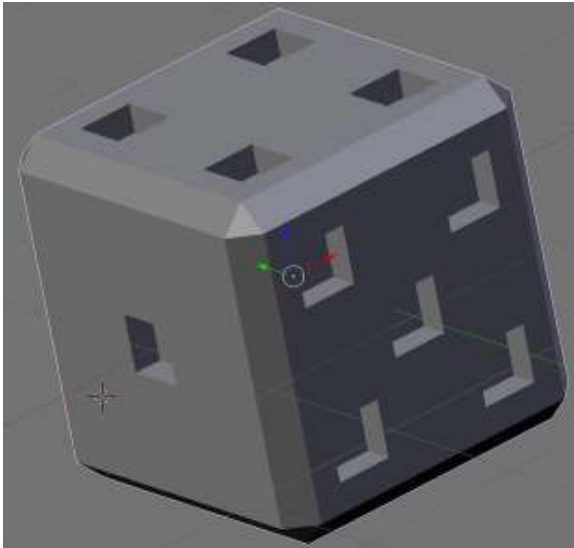


Figure 273

47.5. Subsurfing and smoothing



Figure 274 The *Editing* panel icon, *Modifiers* panel and *Set Smooth* button

Leave the *Edit Mode* for the *Object Mode* (**[TAB]**) and select the *Editing* panel in the *Buttons* window (or press **[F9]**).

To turn on subsurfing, click on *Add Modifier* in the *Modifiers* panel (if you can't see the button, make sure the tab is selected) and select *Subsurf* from the popup menu.

The default value for *Levels* is 1 and for *Render Levels* is 2. For a nicer preview, change *Levels* from 1 to 2 by clicking once on the right arrow, and click on *Set Smooth* in the *Links and Materials* panel.

The cube will now look like this (*Front view*):

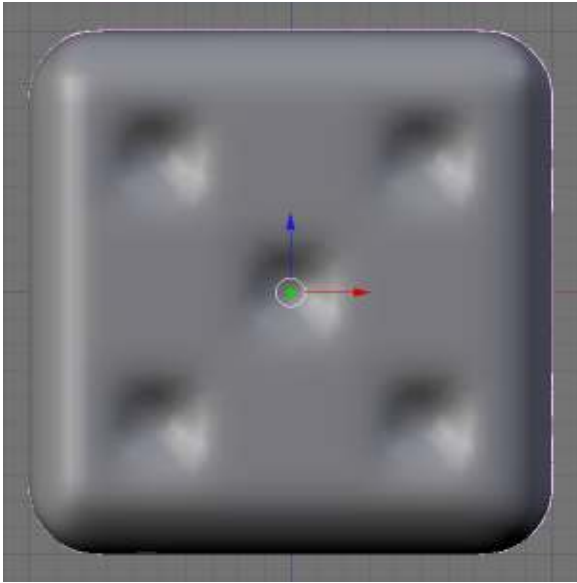


Figure 275 *method 1 & method 2*

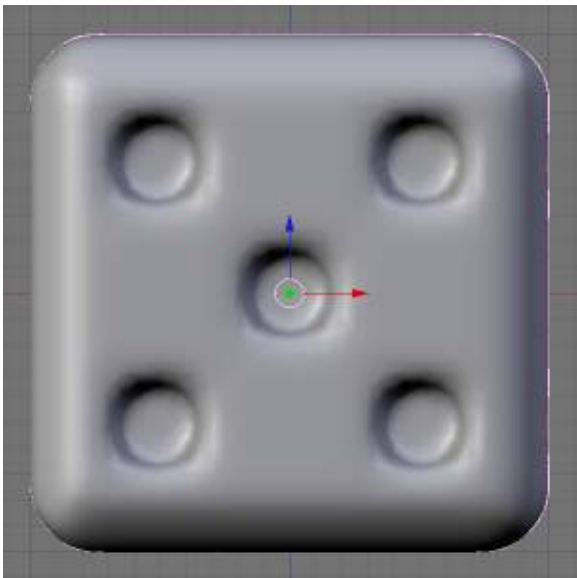


Figure 276 *method 3*

47.5.1. Subsurface creasing

On a die, the edges around the spots are usually sharp so we'll use subsurface creasing to do that.

Go back to *Edit Mode* (`[TAB]` once or twice), and *Edge select mode* (`[CTRL] + [TAB] [2]`) either select every spot perimeter as shown below (`[RMB]` for the first one, `[SHIFT] + [RMB]` for the others), or use circle select as described next

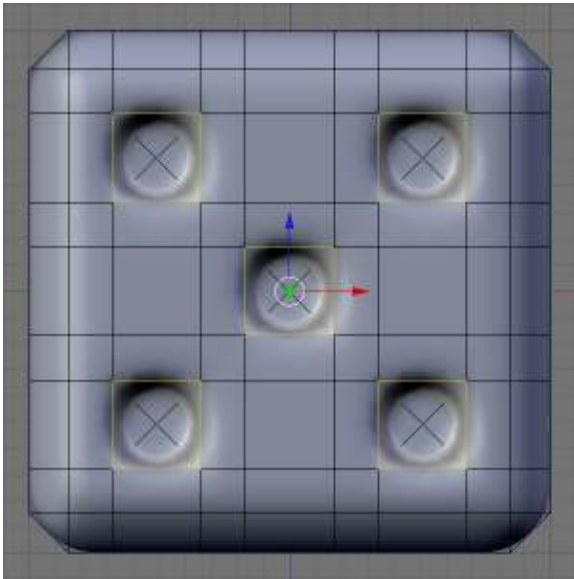


Figure 277

Using circle select

You should be in **Edit Mode**, be in "Edge select mode" (`CTRL` + `TAB` `2`), and have **occlude background geometry** turned on.

`A` once or twice to be sure everything is deselected.

`B` , `B` to TURN ON circle select mode.

Noob note: In Blender 2.5x circle select mode can be used by pressing `C` once.

`MMW` roll (don't press) middle mouse wheel to size the circle small enough to select single edges.

`LMB` about twenty times, as there are 4 edges around each of 5 "dots", so you will have to click 20 times with the left mouse button (once on each edge) to select them all. If you select an extra unwanted edge anywhere Then you can just click the `MMB` (My wheel rolls or presses in).

`RMB` to TURN OFF circle select mode.

Creasing the edges

Now move mouse pointer crosshairs (don't press any button here) near center of drawing.

`SHIFT` + `E` to turn on Edge Crease mode. The bottom bar changes to a "Crease" number.

Now move mouse pointer crosshairs (don't press any button here) away from center of drawing. Watch the Crease number change as you pull out. It maxes out at 1.000.

`LMB` When you get the crease you want the left mouse button locks it in place.

`A` once, or twice, to deselect every thing.

Finished on this side. View a different side and do it all again.

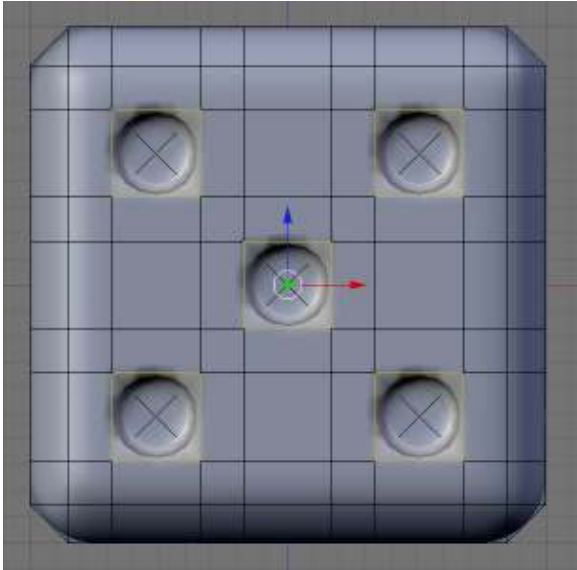



Figure 278

Note: the  shows an information box (with the perimeter edges still selected). The last line says: *Median Crease W: 1.000*, if the value is less than 1.000 it means that an edge has been missed. You can update the crease values by clicking on the arrows. Click on the X in the top right corner to dismiss the dialog.

Noob Note: *setting the value of the subsurf creasing affects the subsurf modifier. You will not actually see the creasing effect unless the object has the subsurf modifier active. Try adding the subsurf modifier to the object then set the creasing to one so that you can see the effect it actually has on the object.*

Now crease the spots on the five other sides of the die. You should end up with:

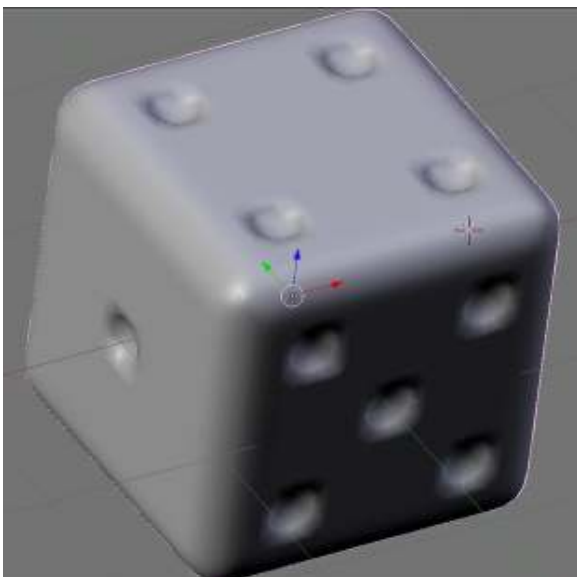


Figure 279 *method 1 & method 2*

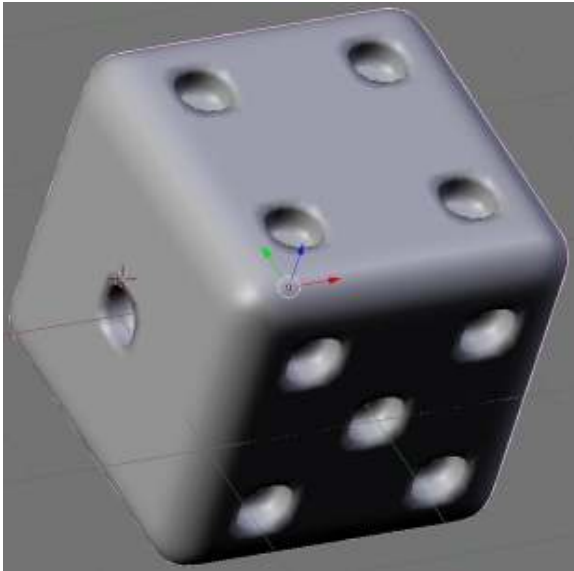


Figure 280 *method 3*

47.6. Colouring the spots

47.6.1. Creating a new material

Go to the *Editing* panel (**[F9]**) in the *Buttons Window*.

The *Links and Materials* panel should be showing (in the right column) *DieWhite* and, under it, *1 Mat 1*:

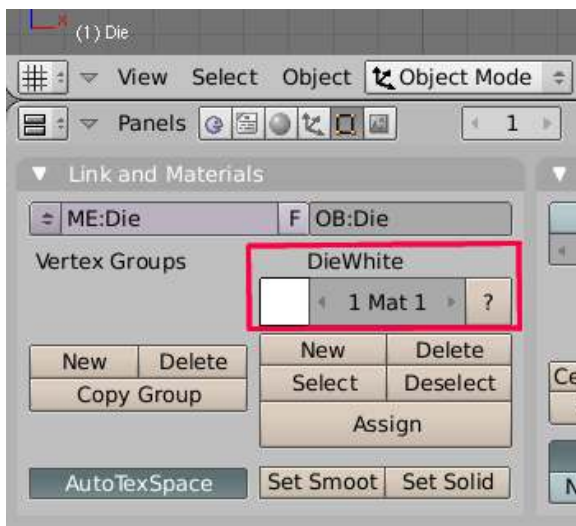


Figure 281

To add another material, click on the *New* button (right under *1 Mat 1*). This label changes to *2 Mat 2*, and *DieWhite* changes to *DieWhite.001*.

Note: if your Blender version is older than 2.44, there's an additional step. Go to the *Links and Pipeline* panel (*Shading* or **F5**), click on the 2 button right after the *MA:DieWhite* input and choose *Single user* in the popup menu.

user request: please provide instructions for 2.45, 'Links and materials' panel has been replaced with 'Links and Pipeline' panel. 'ADD NEW' from dropdown initially containing 'MA: Material' does not increment the '1 Mat 1' fields leaving no clear way to add more than one material to an object. --

Answer from another user: *You're looking at the Shading panel (F5), you need to switch to the Editing panel (F9), there you'll find the buttons just as described here.*

Set the colour for this new material: click on the white square on the left of *2 Mat 2* (in the *Editing* panels) or on the left of *Col* (in the *Shading* panel). To get the same colours as in this tutorial, choose black. Press **ENTER** to confirm your choice.

In the *Links and Pipeline* panel (*Shading* or **F5**), change the material name from *DieWhite.001* to *SpotBlack*.

47.6.2. Selecting the spot faces

Now go into *Edit Mode*, choose *Face select mode* and make sure *Limit selection to visible* is on.

Version 2.46 The "Limit Selection To Visible", is now "Occlude Background Geometry". It remains the button to the right of "Face Select"

Starting again with *Front view* (**NUM1**) use circle select (**B**, **B** — use **SCROLL** to reduce the size of the circle to less than the spot size so you don't select the outside faces) to select all the inside faces of each spot.

Press **RMB** to finish, select the *Back view* (**CTRL** + **NUM1**), use circle select again and select the spot faces there, do the same with the other four die faces.

Note: *you can't change to a different face without stopping the circle select tool (RMB). You will have to click RMB, change to another face, and start circle select again (B, B).*

This is what you should see when you have selected all the spots:

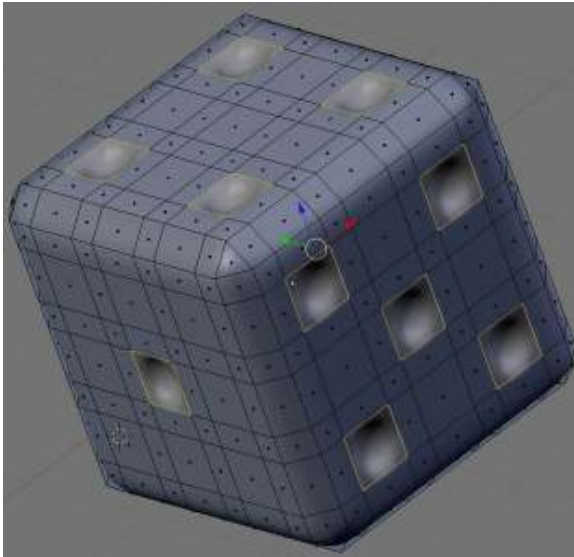


Figure 282 *Alt-M*

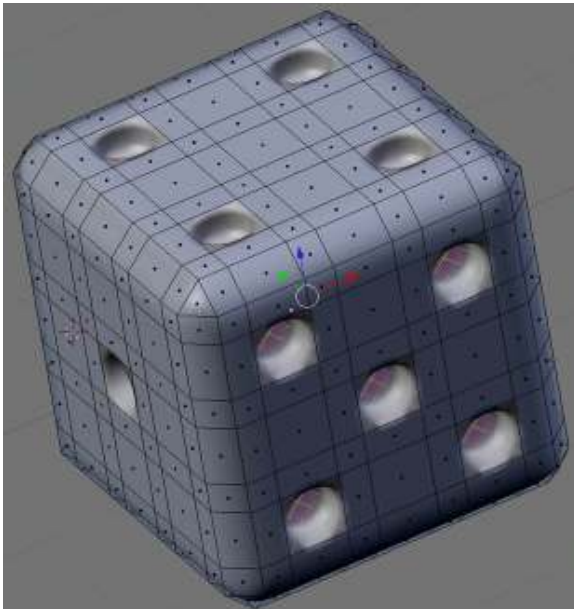


Figure 283 *Ctrl-T*

The selected faces internal to the spots are more visible in the *Alt-M* version as the spots created in *Ctrl-T* are more cylindrical; the *Alt-M* spots look more like hemispheres.

47.6.3. Assigning the new material to the spots

Now that all the spot faces are selected, you can assign the *SpotBlack* material to them.

Go back to the *Editing* panels (**F9**) and, in the *Links and Materials* panel make sure the material *SpotBlack* is selected. If it isn't, you can select it by clicking on the right arrow or the *l* on the right

of 2 *Mat 1*, which should update to 2 *Mat 2*. Now click on the wide *Assign* button below and the spots will change to black (or the other colour you chose).

Here are the finished results:



Figure 284 *Alt-M*



Figure 285 *Ctrl-T*

47.7. Afterword

Here's an external link to a tutorial showing a completely different way to create a die:

- <http://www.blendernation.com/tutorials/blender-3d-beginner-tutorial-a-die/>

The most efficient and easy tutorial can be viewed at:

- <http://www.youtube.com/watch?v=ZTwPvQbhHdY>

48. Die Another Way

48.1. Video Tutorial

A video tutorial has been created for this chapter in Blender 2.48a.

It is compressed and packaged in the Theora (.ogg) video format and requires a player that is able to decode this codec in order to play it, such as the VLC¹ player which is available as a free download for Windows, Mac, and most Linux distributions. Firefox 3.5 is also able to stream Theora video.

For best results, it is recommended that you save this file to your computer for viewing, rather than streaming it inside a web browser, since it is 1020 x 746 pixels.

Figure 286

48.2. Introduction

In the following tutorial you will be creating a die. You will use:

- polygon mesh
- face loop cutting
- subdivision surfaces
- subdivision creases
- bevel
- set smooth
- multiple materials
- extrusion
- merge vertices
- remove doubles
- constraints

There are two methods to create the circles for the die: subdivide first, or manual sizing. In either case, start with the default Cube.

¹ <http://www.videolan.org/vlc/>

48.3. Subdivide First

The die needs to have a 3x3 matrix for the coloured dots (pips). A quick way to do this is to simply *Subdivide* the cube twice before doing anything else. The disadvantage may be that the spaces for the pips may not be exactly the size that you want. If not, see the next section: Manual Sizing of Pips.

48.4. Manual Sizing of Pips

48.4.1. Step 1

Hit tab to go into edit-mode and select all faces to prevent bevel messing up normals. Hit **WKEY** → *Bevel, Recursion* → 1 (you'll see why later) then choose bevel size (hit spacebar for manual input). Bevel of 0.150 is ok.

Note: If you have chosen to subdivide the die twice, jump to Section "Creating Pips" and put bevel of 0.17 in order to have pip's edges length 0.34

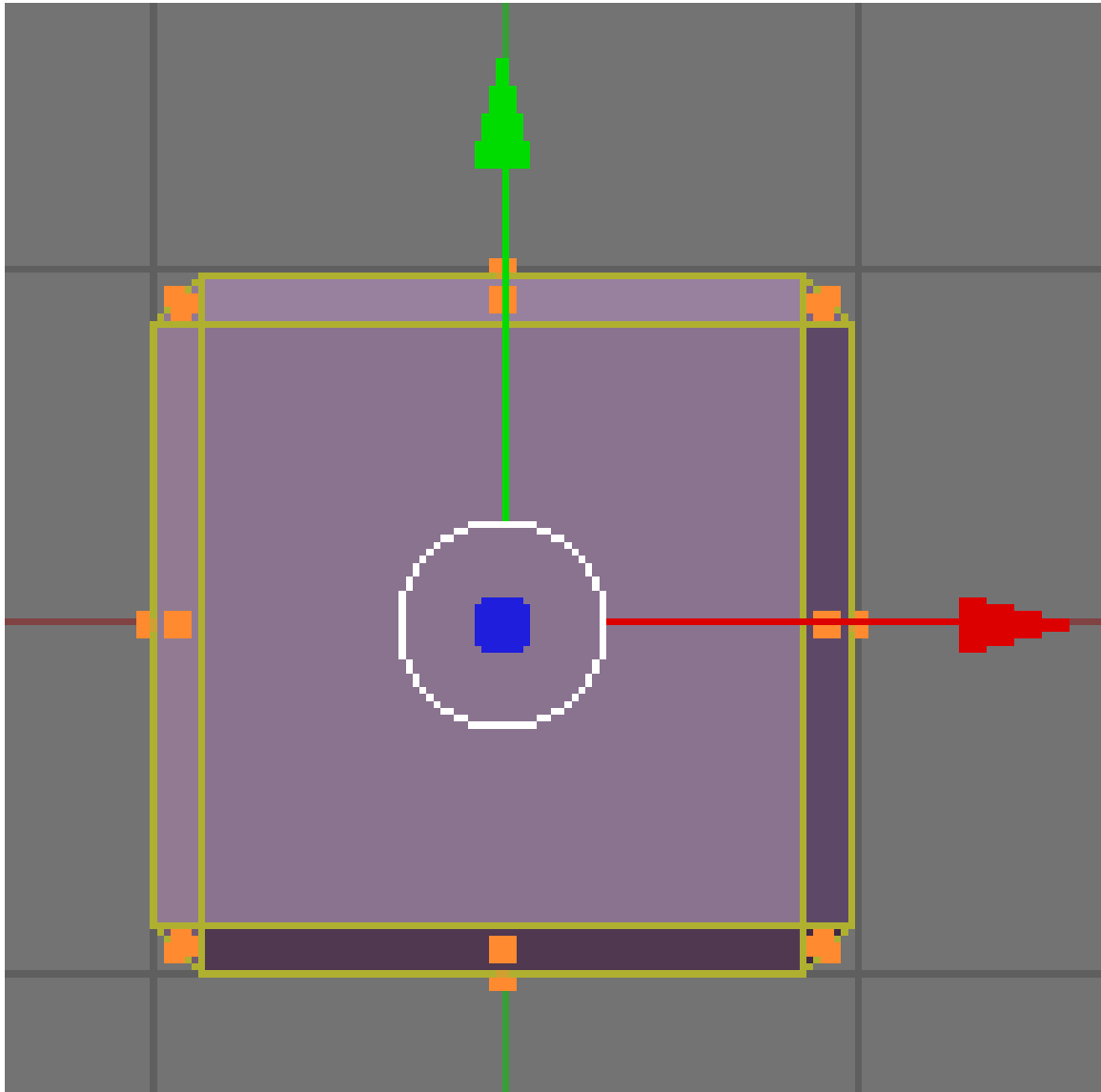


Figure 287

48.4.2. Step 2

In editmode, go to the Editing tab (**F9**) and look at the **Mesh Tools 1** panel (*Mesh Tools More* in some versions). Turn on Edge Length and note the length of one of the sides of the square faces. This should be 1.7 if the above settings were used.

Button "Edge Length" may be outside the screen so you may need to close another set of buttons before you can get to it.

Or, you can use MMB to scroll over to see the Mesh Tools 1 panel (in later versions: Mesh Tools More) with "Edge Length" button on it.

Or, you can zoom in and out in the menu window with CTRL and NUM+ or NUM-

Or, you can press HOME when button window is focused to see all buttons.

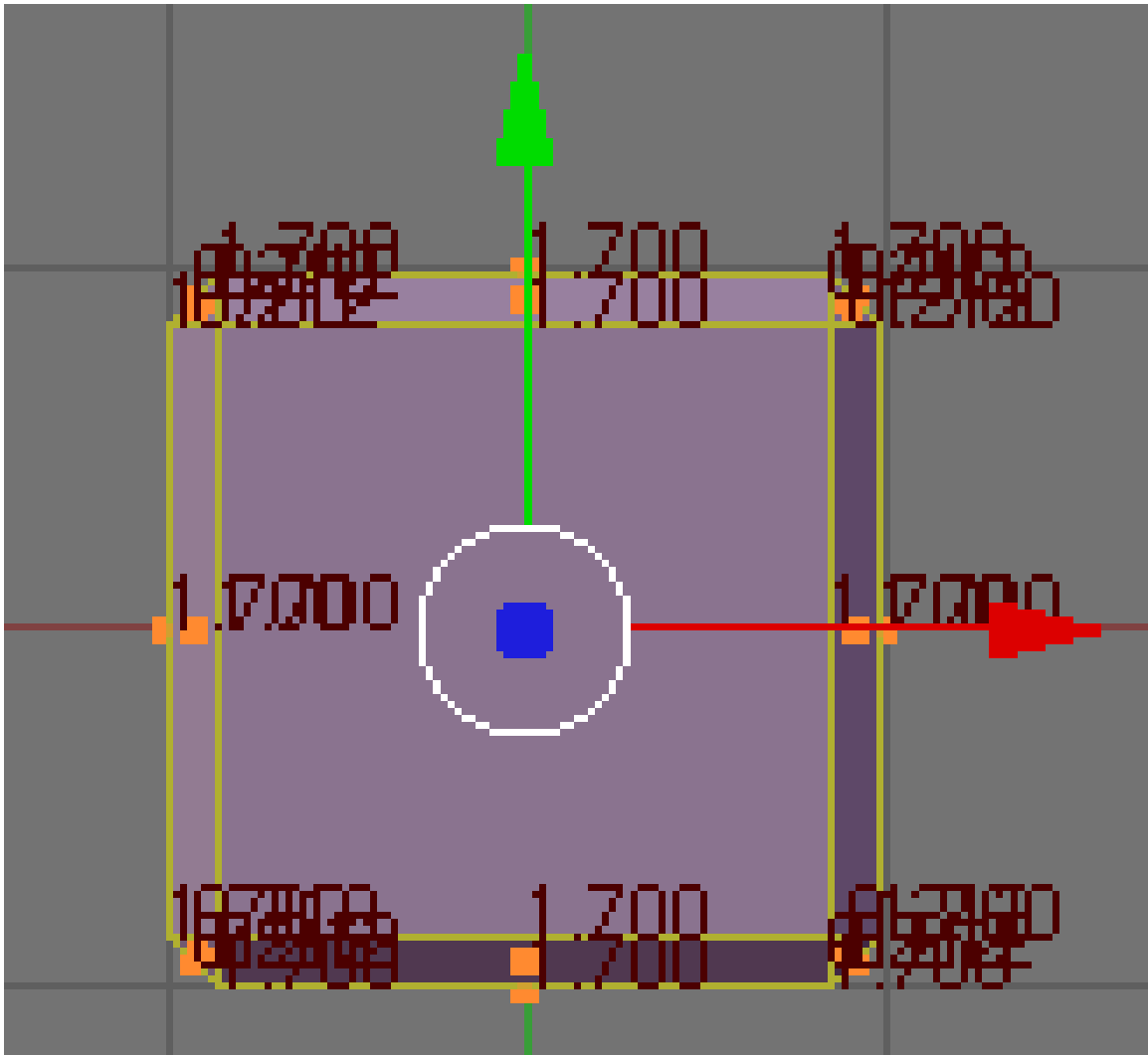


Figure 288

48.4.3. Step 3

A typical die has a grid of 9 possible positions for the pips and the gap between the pips is the pip radius (or half the diameter). So, there are conveniently 10 units on each edge of the square faces, where the gaps use 4 of the units and the 3 pips use two each. This means the gaps are of size $1.7/10 = 0.17$ and the pips $(1.7 \times 2)/10 = 0.34$.

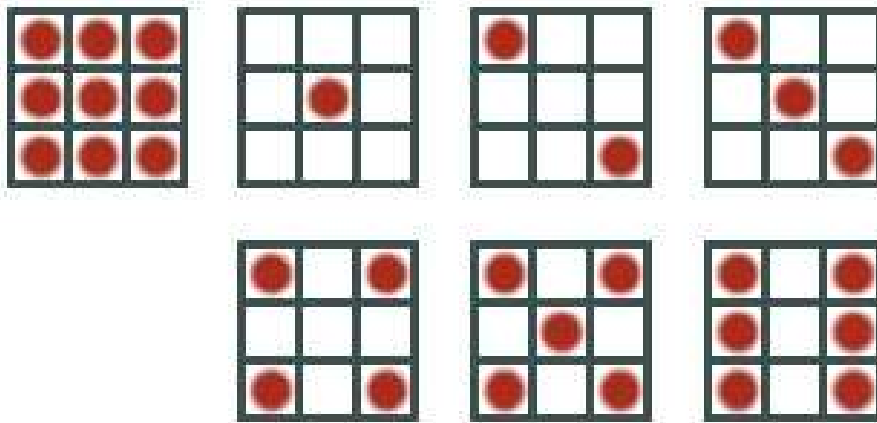


Figure 289

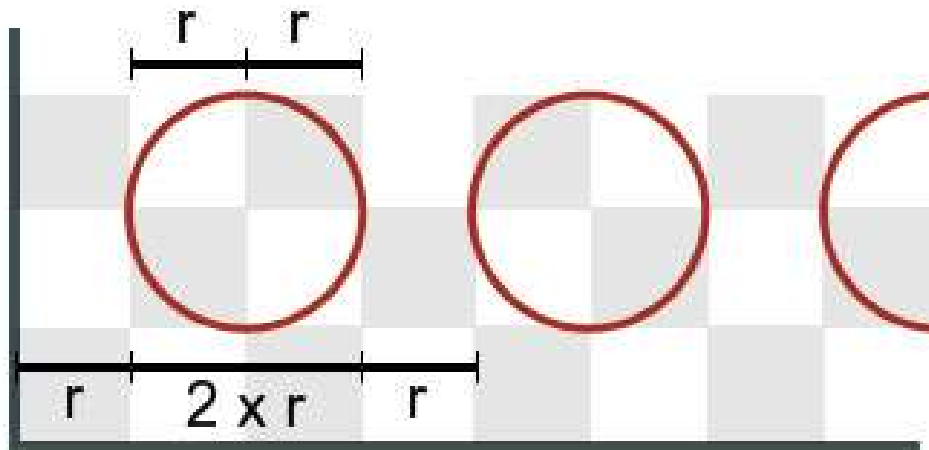


Figure 290

48.4.4. Step 4

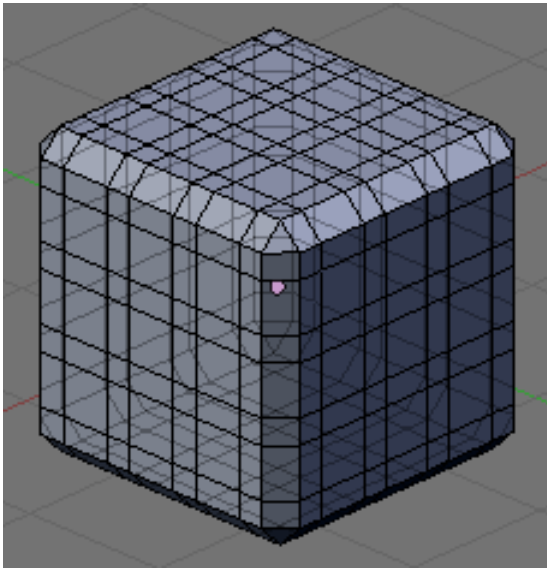


Figure 291

Now it's time to subdivide the surfaces of the die according to the mathematics above. We'll do that using "edge loops" - additional edges you can add to existing objects.

- Select axis aligned view: **NUM1**
- Enter loop cut mode: **CTRL+RKEY** (OR **KKEY**→**1KEY** OR **KKEY**→*Loop Cut* OR **CTRL+EKEY**→*Loopcut* OR **CTRL+EKEY**→**NUM5**)
- Select loops' placement: move the mouse around until you see a purple line going the right direction.
- Enter the number of loops: **9KEY** (OR **NUM9** OR **SCROLL** up 9 times OR **NUM+** 8 times)
- Add the loop: **LMB** (OR **ENTER** etc.) on one of the big faces

Noob: What's "the right direction?" What should this thing look like after applying the loop/cut? HELP!

Another Noob: It sounds like you have version 2.44 or later. You should do the tutorial "Die Easy"(seriously, it is).

Now we just have to get rid of the 2nd, 5th and 8th loops to make the undivided spaces for the marks.

- Select edge select or vertex select: **CTRL+TAB**→**NUM2** (OR **CTRL+TAB**→*Edges* OR **CTRL+TAB**→**NUM1** etc.)
- Deselect all edges with **AKEY**
- Choose a loop to remove (using the **BKEY** to enter box selection mode and drawing a box around the one you want; this will get the whole loop, all the way around the cube). Spoiler: you can also use **Alt-RMB** on an edge to select a loop (or select the edge and click *Select* --> *Edge loop*).

Noob: It seems that I could not remove multi loops at the same time, since the error message kept bumping out. Another Noob: That happened to me when I tried to remove them all at once, but it worked fine when I removed one at a time.

- Remove loop: **XKEY**→7 (OR **XKEY**→*Edge Loop*)

Change views with NUM3 and NUM7 and repeat steps as necessary. When you're done, your die should look like the one pictured to the right.

48.5. Creating Pips

The die needs the pips added. Everyone knows how the pips on a die look, right?

48.5.1. Extrude and Merge

(Note: This is Step 10) Select one of the faces where a pip would go and extrude the face by hitting **EKEY** and then **ESC**. Do not click after hitting **EKEY**. This actually replaces the first face with another one even though it looks like nothing has happened. Merge the second face by using **ALT + MKEY** to merge the 4 corners into the centre. It will tell you *Removed 3 vertices*. [User Note: To clarify, pressing ESC when extruding does in fact extrude the face as per usual, but by a distance of zero. This creates four new, infinitely narrow faces around the original square face. These four faces then get 'dragged' into the middle when the four vertices of the original square are merged into one. Test this by extruding by say, 0.01 (instead of pressing ESC), and you'll see the result is almost the same.][User Note: on my computer hitting ESC after the EKEY seemed to cancel the extrude and to get the correct result I had to hit the "ENTER" key after the EKEY] You should get the following:

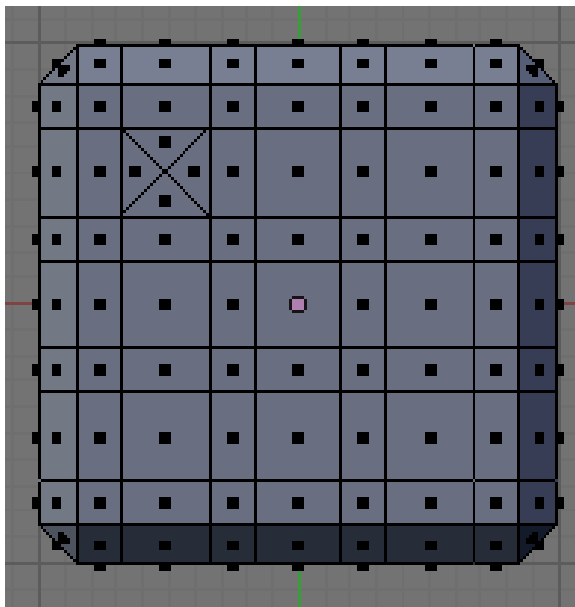


Figure 292

Do this for the configuration of the dots on that side. So for example, 5 would look like this:

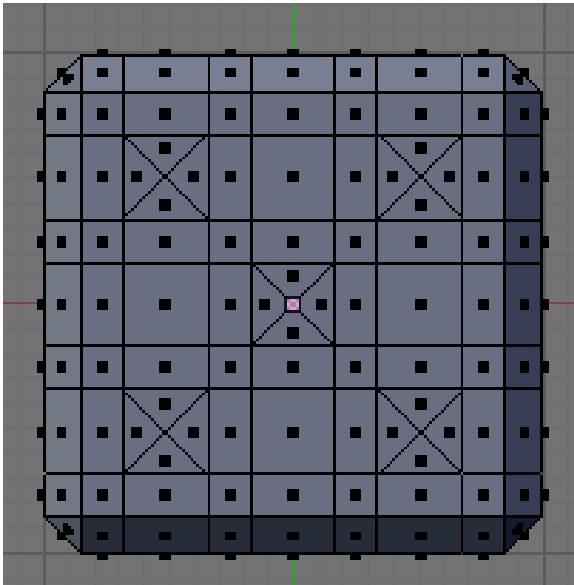


Figure 293

Notes :

- You could create this pip spot on all 9 spots and copy this side of the die to the other six. The amount of time spent doing all of that may be just as long as doing each side individually. You would need to delete the other 5 faces, copy the dented face 5 times, place each face precisely by rotating and moving, and remove doubles.
- (User comment) I accidentally selected some pip faces from the opposite side of the die (the side behind the side I was looking at). To prevent this, I selected *Limit selection to visible* (*Occlude background geometry* in 2.47)

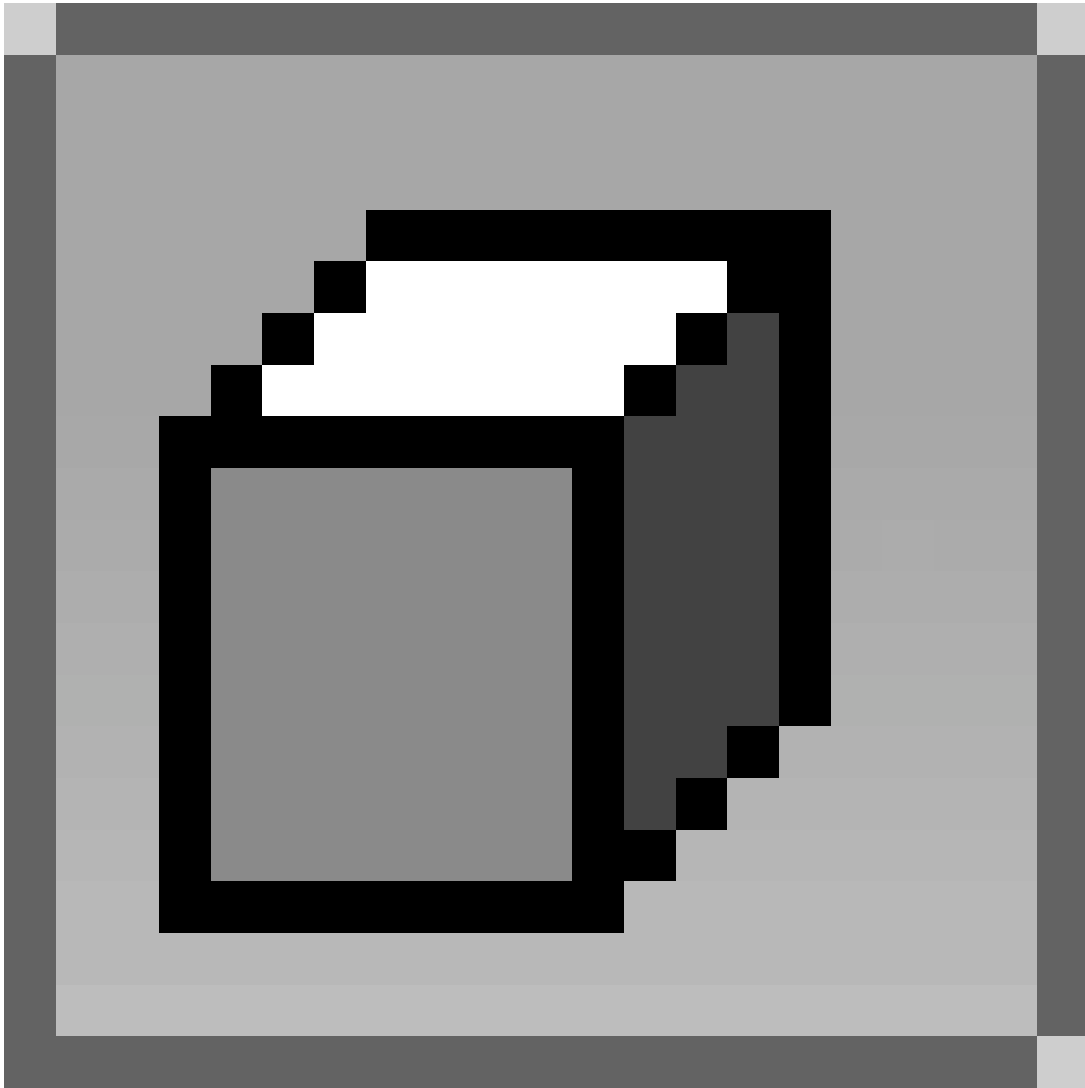


Figure 294

: which should be the second button from the right in the header of the *3D View*.

- **Noob, 19th Oct 2008:** You can save time by selecting all the 'pip faces' and extruding them simultaneously, **ESC**-ing immediately after you do so, like above. You'll still need to merge the four corners of each extrusion one-by-one, though, or you'll get some odd results.
- **Noob, 28th Dec 2008:** You can save more time by extruding all faces simultaneously, as above, and then selecting "Collapse" rather than "At Center" while merging.
- **Noob, 02/02/09:** I tried with the "Collapse" trick as well but all I got were just black squares.
- **Noob, 19/02/09:** I got the black squares as well, but it is just a display problem. Press *Tab* twice (going to *Object mode* and back to *Edit mode*) and they are gone.

48.5.2. Create Pips

(Note: This is Step 11) Select one of the edges of the pips to check the size is 0.34.

Remember the pip radius was 0.17. We need to use this value to lower the centre point of the pips. Select all the 5 centre points at once to save time and move them inwards by 0.17. The side I put the 5 pips on here was the top so I move the vertices inwards by pressing **GKEY**, **ZKEY**, -0.17 and hitting **ENTER**. I then get this:

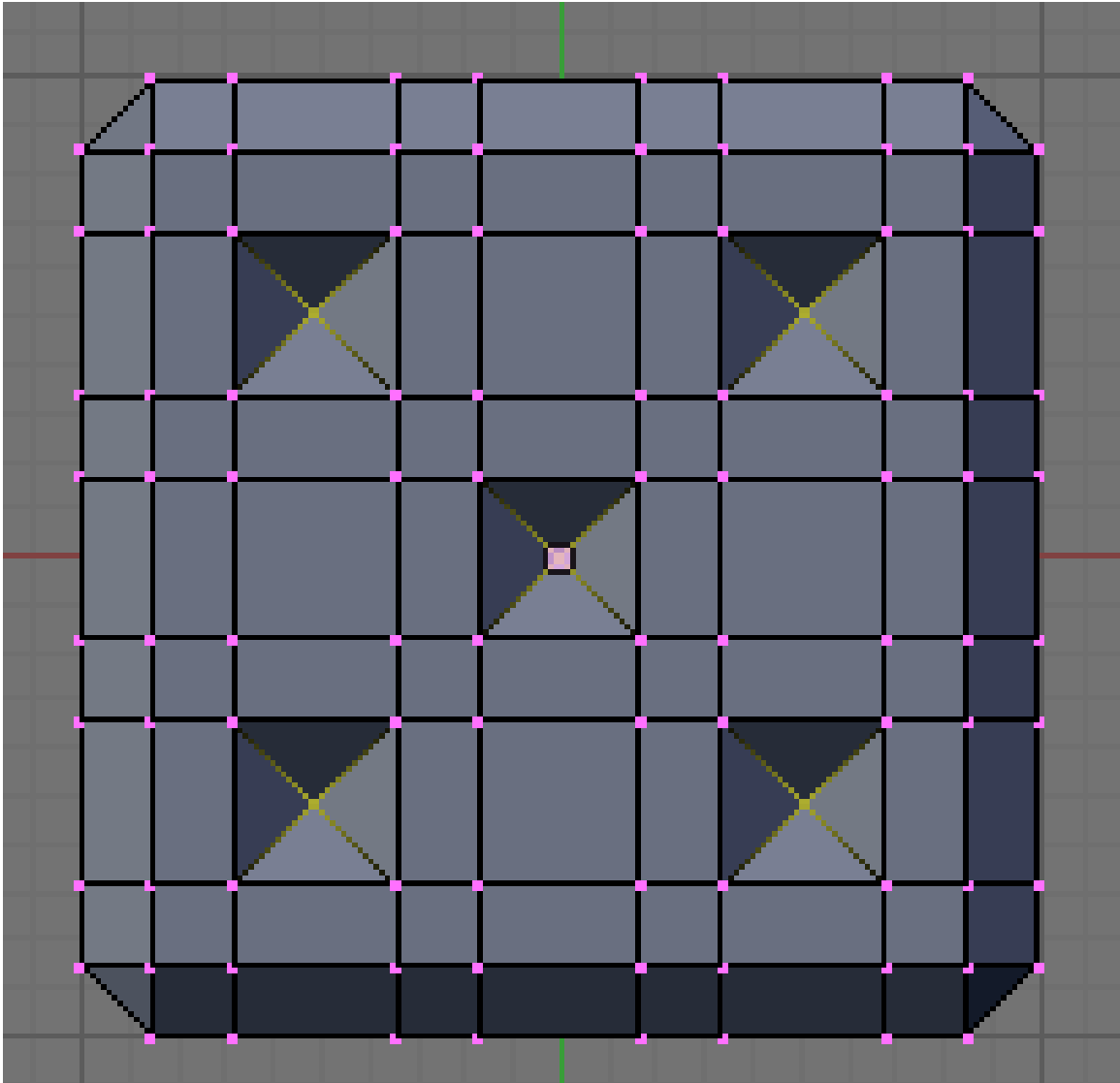


Figure 295

(user comment) According to step 4 we are still in front-view (**NUM1**), but then the **ZKEY** modification gives undesirable result, changing view to top (**NUM7**) does the trick! This applies to blender version 2.44.

- (response) Actually, at the end of step 4 it states to change to side and top view (**NUM3** and **NUM7**) as necessary, so really there is no official view the tutorial left the user in. Also, in this step, the writer mentions they put 5 pips on the top.
- (user comment) using **YKEY** instead of **ZKEY** is also fine.

- (user comment) Depending on which axis it's supposed to be moved, use the **ZKEY**, **XKEY** and **YKEY** after the **GKEY** accordingly.
- (user comment) Pressing **ZKEY** twice should move them along the normal, which should work no matter which faces are selected.
- (user comment) I noticed a shortcut: When you extrude, extrude by -0.17 and then do the Merge -> Collapse. Then the point is already inside.
- (user comment) I just noticed something different. If you extrude and collapse, after applying the subsurf, you got sharp edges of the pips than the pips I got in this tutorial way.
- (user comment) In blender 2.49a there is an issue where when "Occlude background geometry" is on, sometimes you'll click a vertex and have a far away vertex clicked instead. Rotating the camera usually helps, and if you find your vertex isn't selected, but you know you clicked, clicking again will deselect whatever vertex you selected. Also a big issue with the extrusion - as with escape, right clicking to exit extrusion also creates an "unseen extrusion" . The "unseen extrusion" can be noticed - there's black dots in the middle of your square edges. This is caused by the 0-width faces being selectable (and therefore having a dot in the middle). I had to completely do-over my model because Remove Doubles _did not_ remove the extra vertices. This became noticeable when I would extrude and it would only extrude vertices.
- (user comment): Don't collapse the vertices! In that case, the pits will be rendered badly. Collapsing removes four vertices, merging at the center removes three vertices --> so there is a difference!
- (user comment): You can just have the pivot be Individual Centers and scale to zero, then you can remove doubles and it works out fine. That's the fastest way I can think of doing this.

48.5.3. Smooth Out

TAB out of Edit Mode. If you haven't done this already, hit Set Smooth in the Editing panel and turn on subdivision surfaces

It should look something like this:



Figure 296



Figure 297

In present versions, you just need to use "Add Modifier" on the Modifiers tab (in Edit-mode), to add a SubSurf modifier. (Or press **SHIFT+OKEY**)

In the image below Levels is set to 3.

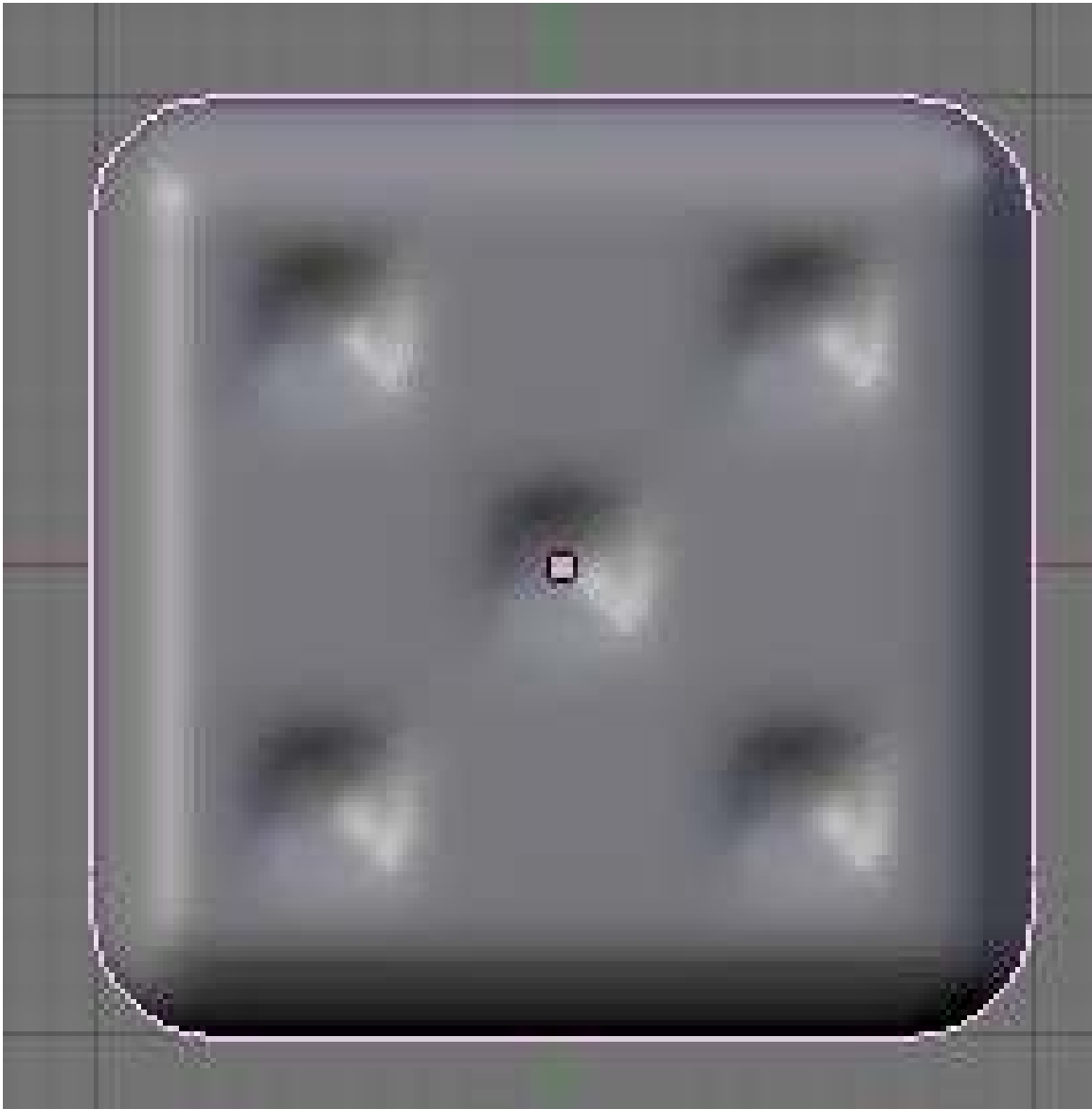


Figure 298

48.5.4. Make Sharp Edges

On a die, the edges of the pips are usually sharp so we'll use subsurface creasing to do that.

Go back into editmode and with the edge select mode on, select all the perimeters of the pips like so (it may help to turn off subsurf for the moment):

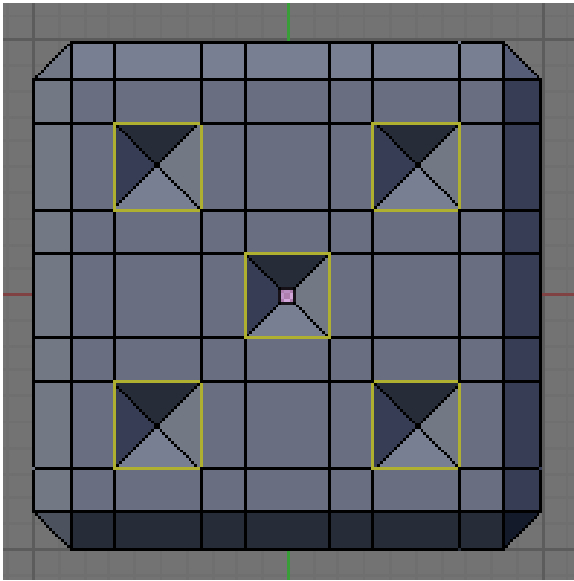


Figure 299

Press **SHIFT+EKEY** to enable creasing and move the mouse until the display says crease is at 1. (to see the effect, you must have the subsurf modifier turned ON). After pressing **SHIFT+EKEY**, you can then set crease values in the information box that you get by pressing **NKEY** when objects are selected. This can be useful to check if all the edges have the right crease because it gives you the average crease value and if it is less than 1, there is an edge wrong.

Newbie Note: Trying to crease all 6 sides of the die at once using **SHIFT+EKEY** and moving the mouse doesn't crease all sides of the die. Better to use **NKEY**, or do one side at a time.

48.5.5. Repeat

Repeat steps 10, 11 and 13 (that is Extrude and Merge², Create Pips³ and Make Sharp Edges⁴) for all the sides of the die. **REMEMBER**, a die is numbered so that opposite sides add up to 7. In my example, that means I put 2 on the bottom etc. Once you finish, if you turn on subdiv level 2, you will get something like this:

[User Note: Do you mean subsurf instead of subdiv?]

- Noob note: On the side with 6 pips, I had a hard time getting the creases to work. It turned out to be because I selected in vertex mode when I selected the edges for creasing, and I tried to crease all 6 pips at once. This put crease values on the short vertical edges between the pips, which messed things up.

2 Chapter 48.5.1 on page 345

3 Chapter 48.5.2 on page 347

4 Chapter 48.5.4 on page 351

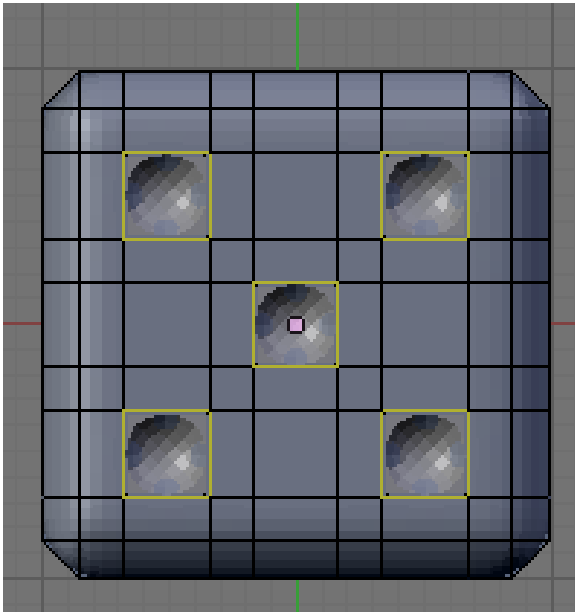


Figure 300

- User Shortcut: To do them all at once with no repeating. However, order of operation has to change to make it work:
 - In Face Select mode, select all pip faces on all sides.
 - **SHIFT+EKEY** to crease the edges. (needs to be done before extruding, since extruding changed the selection)
 - **EKEY** -> **Individual Faces** then type -0.17 to extrude all faces inward.
 - **ALT+MKEY** -> **Collapse** will merge all the extrusions to their respective centers.

(noobie) Some of my pips are square and I tried everything. What should I do? (noob reply) Maybe you had the vertex at the center of pips selected while creasing.

48.5.6. Camera Setup

You can make a test render now to see that the pips are the right size and that the bevel is right. So, turn the subsurf level for the rendering up to 3. To help position the camera so that you centre the die, you can make the camera look at the die by adding a track-to constraint to it. I prefer to track an empty though, because it is more flexible.

Make an empty by going into top down view (**NUM7**) and hitting **SPACE** → *Add* → *Empty*. (**Noob note:** If you can't find "empty" in the list, make sure you are in Object mode.) It's always best to go into one of the set orthographic views so as to align new objects to the axes. If you add something misaligned, just go to the object menu then clear/apply > clear rotation (or **ALT+RKEY**). Because the empty was created at the origin, you might not be able to see it as it is inside the die. Hit **ZKEY** to enable wireframe mode and select the empty. Just move it outside the cube until we get the constraint set up.

To add a track-to constraint, select the camera first then **SHIFT+RMB** the empty and press **CTRL+TKEY** and choose "TrackTo Constraint" from the list. Move the empty back inside the die. You can edit constraints in the object tab (**F7**). Add a couple of lamps (both intensity 1) to get the scene like this or feel free to experiment with a more advanced lighting setup:

Another way to position the camera is by selecting it and then looking through it as you move it. Look through the camera by pressing **NUM0**. Use the **GKEY** to pan across and rotate around the local axes of the camera by pressing say **RKEY,XKEY,XKEY** to rotate in X-axis. To zoom in and out press **GKEY, ZKEY, ZKEY** and then move your mouse forwards or backwards. Another useful keystroke (for pre-2.43 especially) to know is that when you are in camera view, pressing **Gkey** and then **MMB**, movement will be constrained to the way you are facing. The **mouse wheel** zoom moves your view towards and away from the camera, without actually moving the position of the camera.

You can also move the Camera in free "Fly" mode by going into the Camera view (press **NUM0**) and then **Shift+F**. Now you can "Fly" through the scene and use this setup the camera angle. Make sure that you key the flying velocity very low by using the scroll wheels or **-/+** buttons or the camera will be simply out of control.

48.5.7. Render

To render, set the size of the image you want. 800x600 is a decent size so put these settings in the format panel in the Scene tab (**F10**). In the render panel, make sure 100% is selected. If it's 50%, the render will come out as 400x300. For preview renders, don't turn on OSA, which is anti-aliasing because it slows your renders down significantly. Try to only use it for a final render.

Another important point is to set the image format. This is done in the format panel. The listbox has a number of image types. I find that png is generally the best because it is lossless and offers the highest compression among the lossless formats. It also supports an alpha channel for transparency. When rendering an animation, it is better to render as an image sequence than as a movie because it is easier to edit these and repair broken frames. Quicktime supports loading of image sequences and you can save as a movie using a wide range of compression formats.

To save the render, go to the **file menu** → **save image** (Or press **F3**) and type in the full name of the image including the extension e.g. die.png.

The output should now be looking something like this:

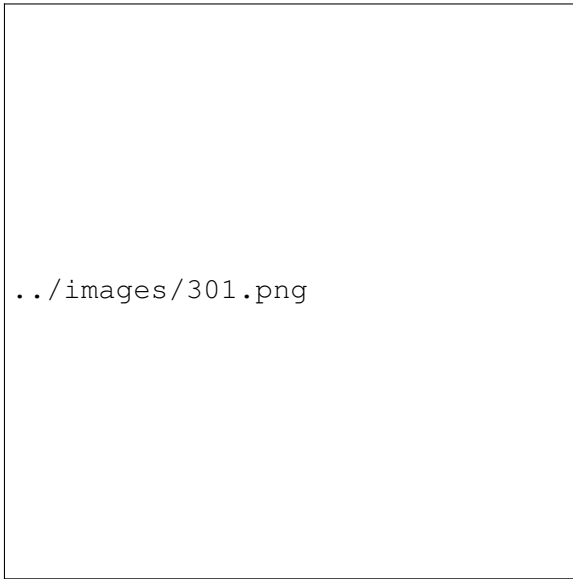


Figure 301



Figure 302

[user note] To get such a nice render I had to bump the render level of subsurf up to 5, otherwise I get artifacts around the pips

[user reply] If OSA is on for the final render, you'll get a nice render result as shown by the above figure with subsurf level 3.

[user note] If you set smooth on the whole shape, you'll get artifacts around the circles.

[user reply] To avoid artifacts around the circles, you'll have to turn on **Auto Smooth**

48.6. Color

48.6.1. Multiple Materials

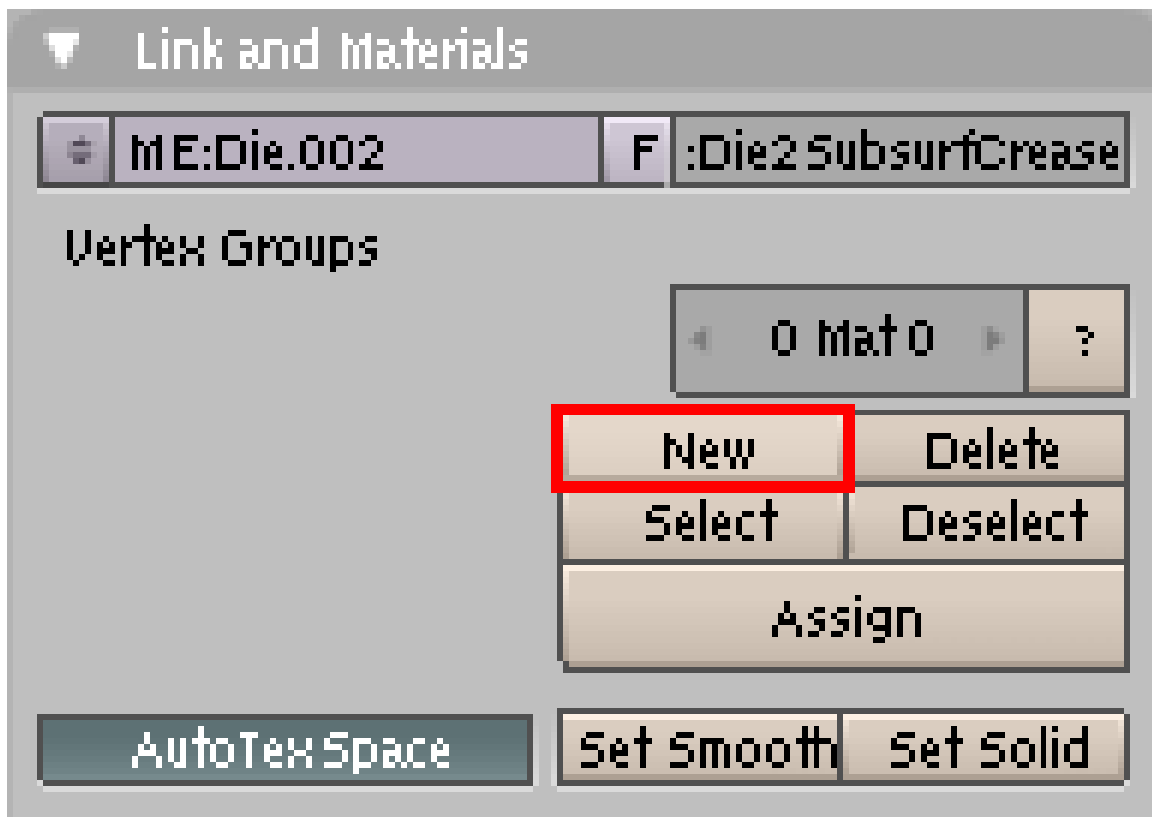


Figure 303

..

To give it some colour, we will need to use multiple materials because a typical die has pips that are a different colour from the die itself.

In the *Buttons Window*

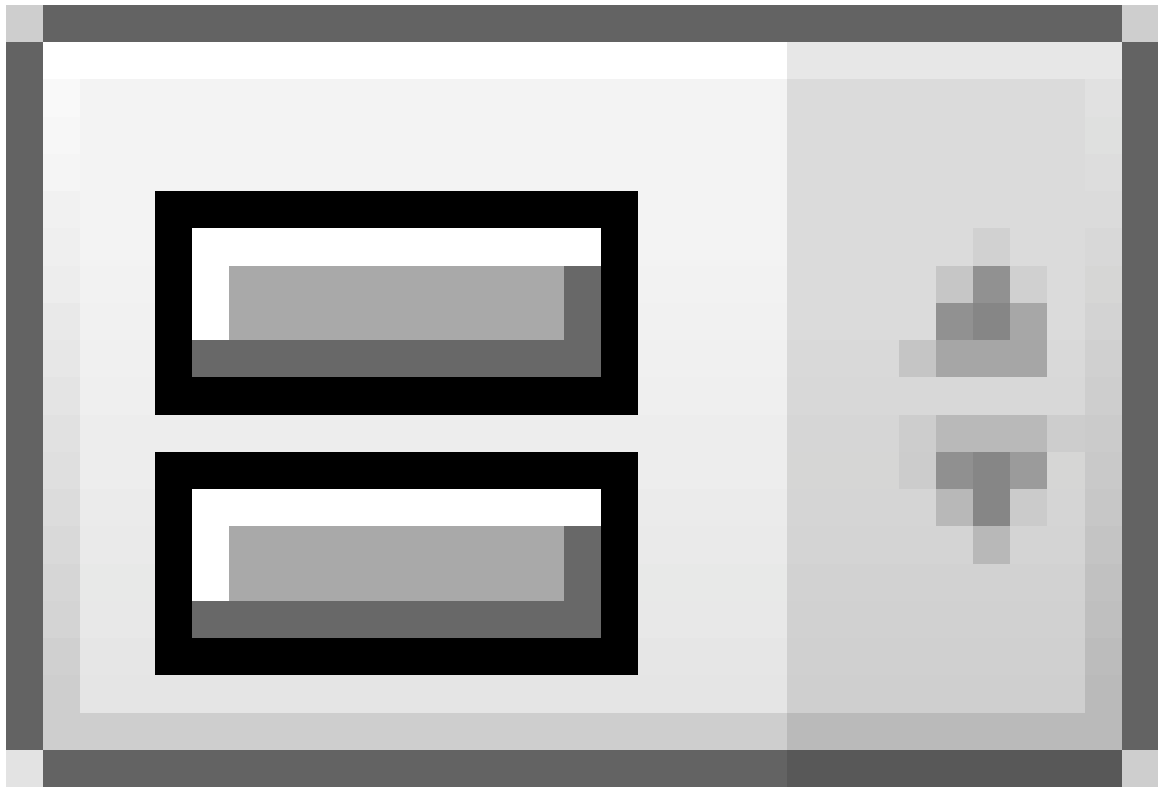


Figure 304

go to the *Editing* panel

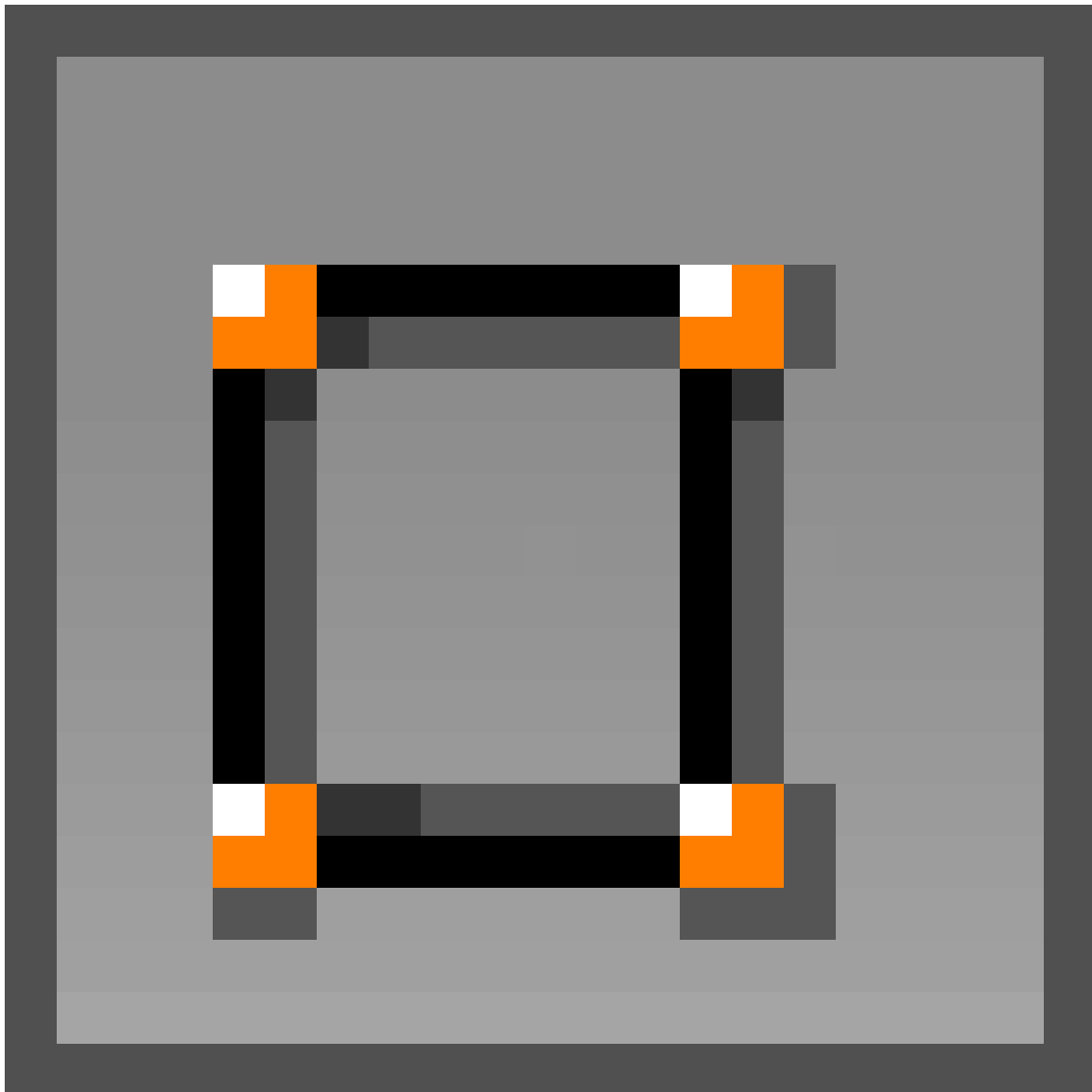


Figure 305

(F9) again and make sure the die is selected. In the *Links and Materials* subpanel there is a section for materials (the right; the left one is for vertex groups) and the box left of the question mark should read "0 Mat 0" (the first number is the number of material links for this object; the second number is the number of the currently selected material link).

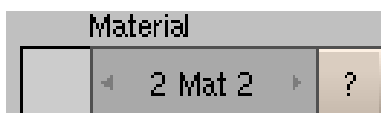


Figure 306

The die may have more than zero materials if you had assigned materials to the object already. By pressing the *New* button add enough materials to make 2 in total.



Figure 307

Go back to the *Shading* panel



Figure 308

(F5) and there is a box at the very top of the *Links and Pipelines* subpanel with the number 2 beside it. If there is no such subpanel select *Material buttons*

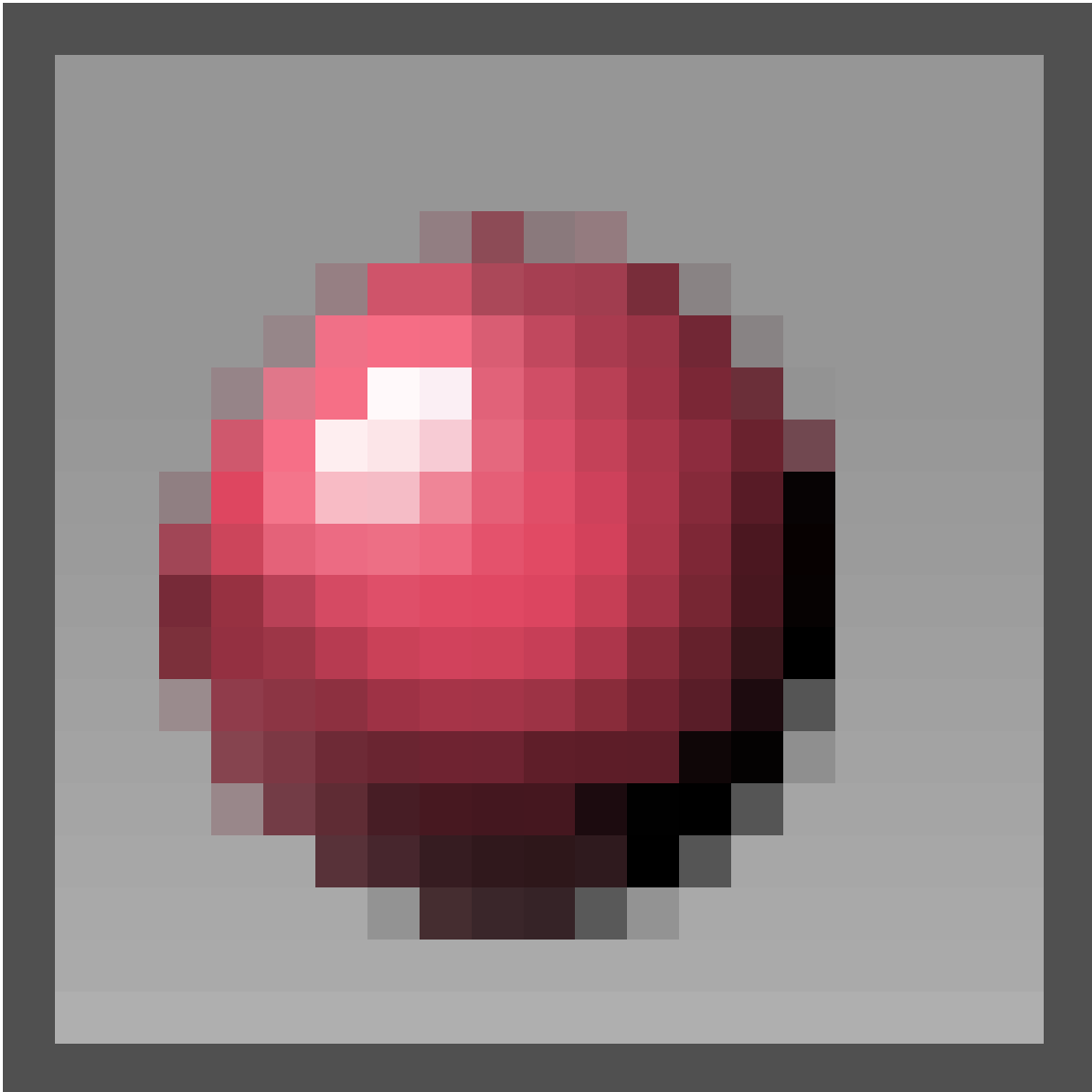


Figure 309

(cycle shading buttons using **F5** too). Click this number and select *Single user* in the dialog to make the two materials you've just created independent. Use the arrows on the left side of the box to switch materials.

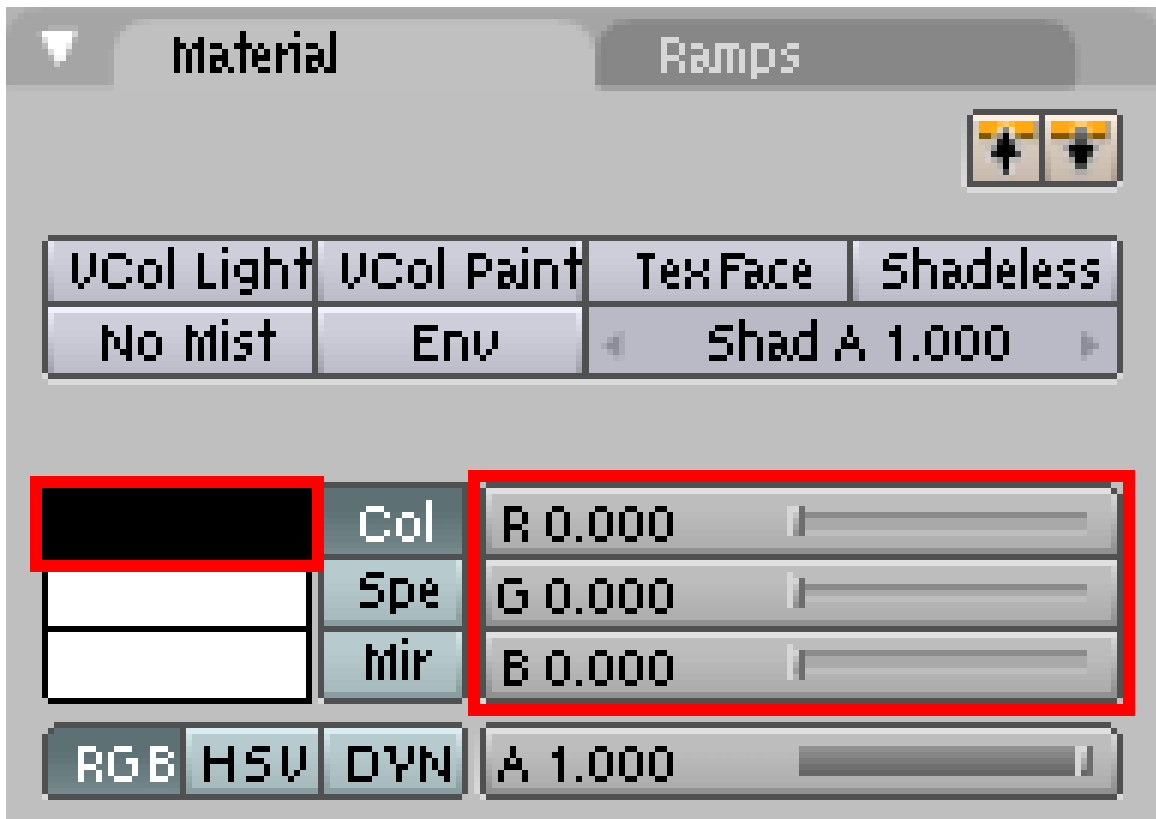


Figure 310

Note: If you do not see a 2 to the right of the material name, that means the material is already a single user material. To change it back you can click the button labelled *F*, but for this example, you do not want to do that.

Note: There should be at least two materials now. One has the materials initial name the other has a number appended to its name (e.g. *Material* and *Material.001*).

Use the *Material* subpanel to make material 1 bright red by just picking red in the colour picker (the rectangle to the left of the *Col* button) or by setting the RGB sliders (right of the *Col* button). Make material 2 white by doing the same. Or pick whatever colour you prefer and material settings.

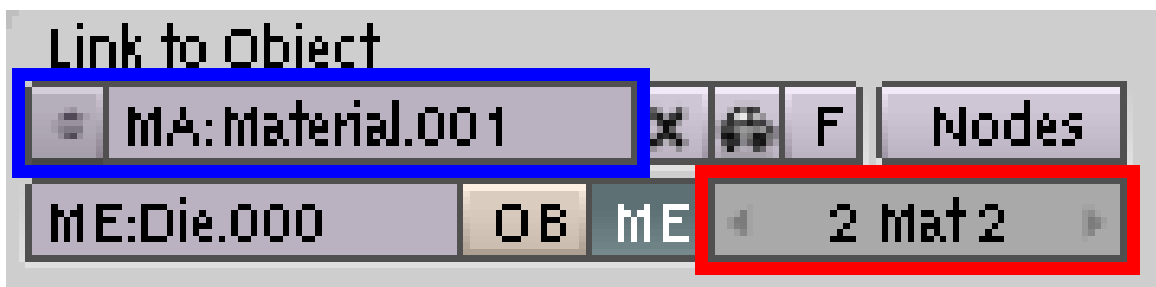


Figure 311

--MSK61⁵ (talk⁶) 11:27, 17 March 2008 (UTC): The colour picker is to the left(not right) of the *Col* button, while the the RGB sliders are to the right(not left) of the *Col* button.

Note: It is possible that the two materials were not automatically linked to the material links of your die. If so use the *Links and Pipeline* subpanel to link the materials to the respective material links. First select the link then the material.

Note from noob: Using white as a colour will not let you spot a change in colour, since the default colour is white. For test purposes I recommend you choose another colour (e.g. blue), so you can avoid getting confused. Note from another noob: Actually (at least in blender 2.49), the default color is grey. To choose a white color, set all colors (R,G,B) up to 1.000.

48.6.2. Assign Materials

These colours need to be assigned to the right parts of the die.

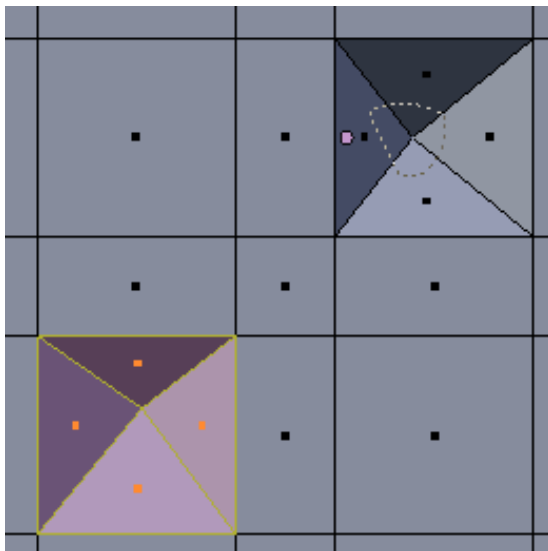


Figure 312

⁵ <http://en.wikibooks.org/wiki/User%3AMSK61>

⁶ <http://en.wikibooks.org/wiki/User%20talk%3AMSK61>

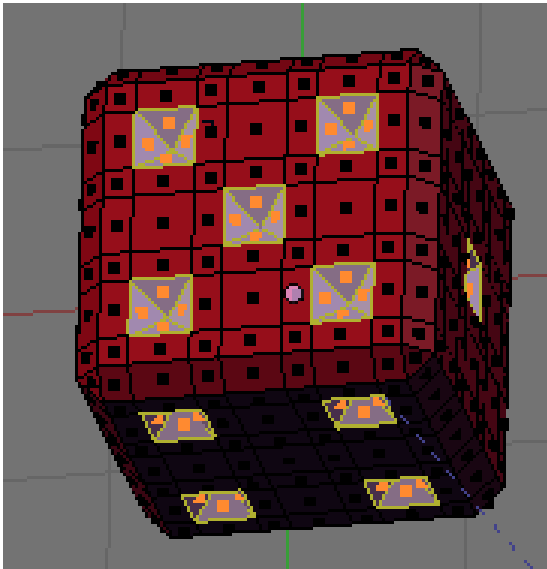


Figure 313

Go into *Edit Mode* and turn off *subsurf* to make selecting easier. Do this in the *Editing* panels (**F9**) *Modifiers* subpanel. Right after the *subsurf* modifiers name there are three buttons (darkgrey). Press the rightmost to deactivate the modifier in the edit mode.

To make the die red

- Select the entire die (AKEY).
- Click the Editing button (F9).
- In the Link and Materials panel, on the right-hand side you will see "2 Mat X" where X is either 1 or 2. Click the left/right arrows until you see the adjoining square to the left turn red. As you click on the arrows, notice that the label above "2 Mat X" changes from "Material" to "Material.001". These are the names of the materials that you created in the Shading (F5)->Links and Pipeline panel.
- Click the Assign button (in the Material column, not the Vertex Groups column). The Assign button associates the selected faces with the selected material.

The entire die should now be red.

To make the pips white

Use the same method as above, but with only the inner faces of the pips selected and with the colour white.

There are a variety of ways to select the inner faces of the pips:

Pip Selection Method 1

Note: Here you may find *Lasso Select* useful.

- Make sure you are in *Edit Mode*



Figure 314

(TAB)

- Select *Face select*

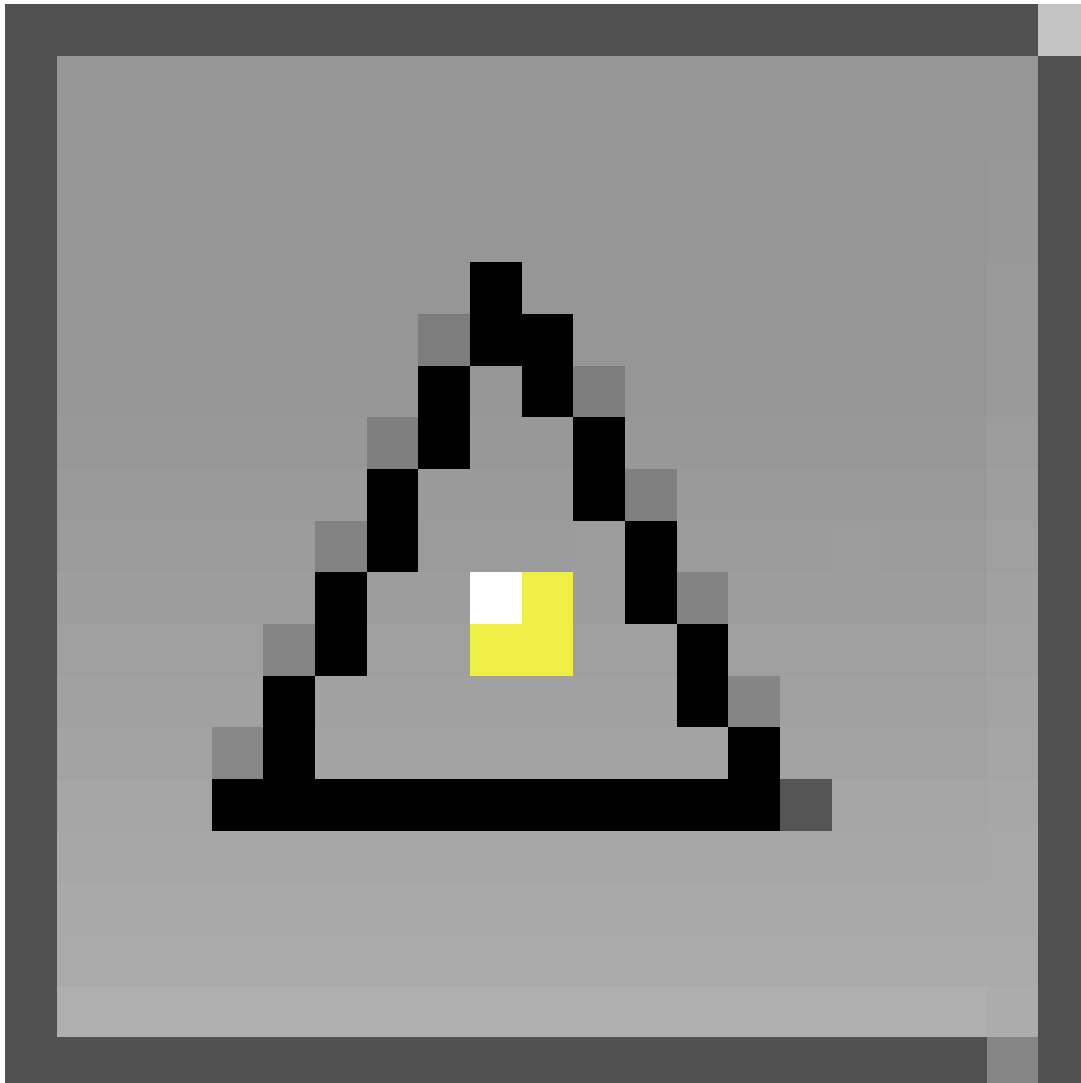


Figure 315

: CTRL+TAB→3KEY

- Select *Limit selection to visible*

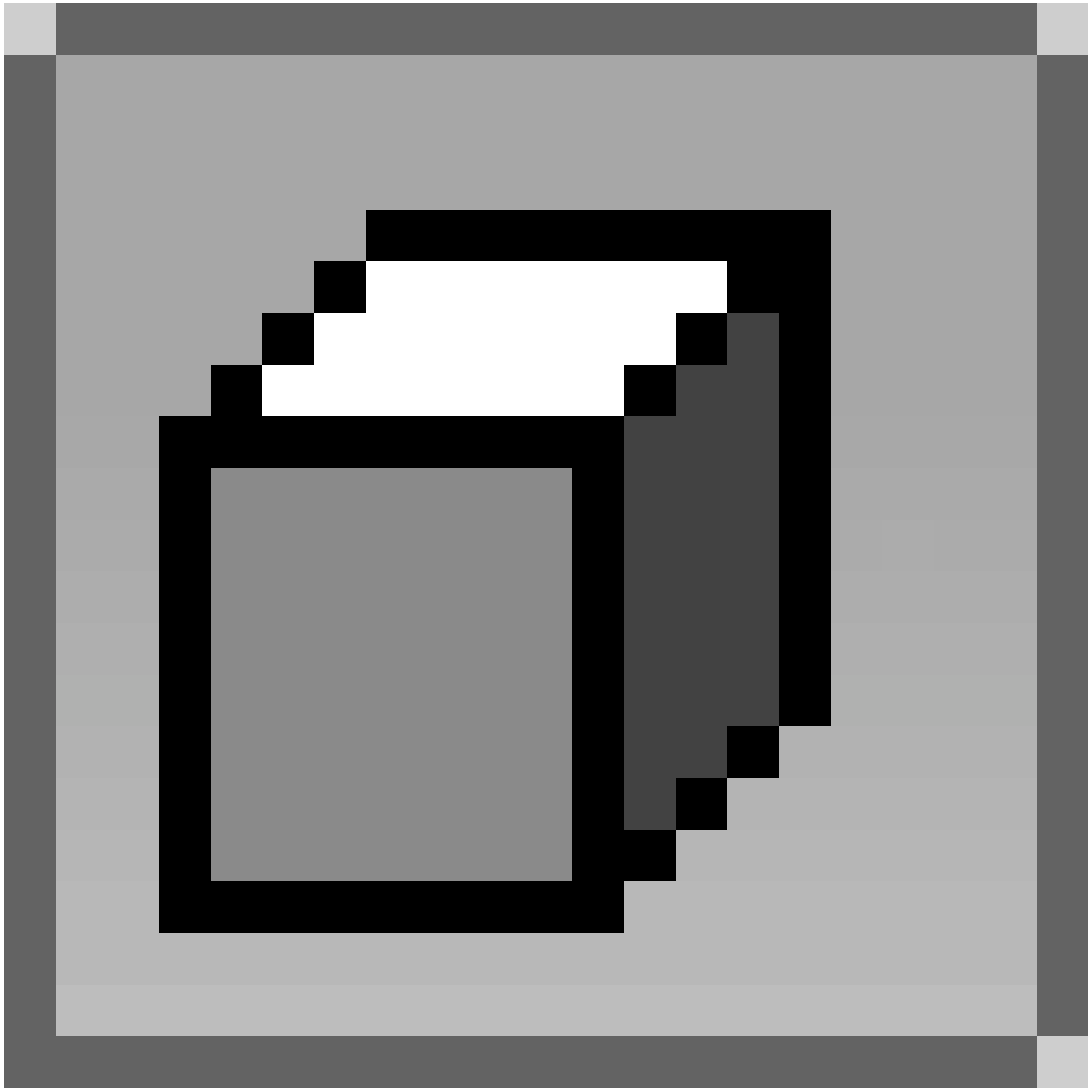


Figure 316

- : should be the second button from the right in the header of the *3D View*
- Go through the axis align views and select the faces:
 - Align view: **NUM1** (**CTRL+NUM1**, **NUM3**, **CTRL+NUM3**, **NUM7**, **CTRL+NUM7**)
 - For each view, lasso select the pips' faces: hold **CTRL** and drag **LMB** around the pips' middle vertex (no need to press **SHIFT**, *Lasso Select* automatically adds the new faces to the previous selection).
 - If all went well you should be able to read *Fa:84-449* in the *User Preferences* right after the Blender version number. ($84 = 4 \cdot (1+2+3+4+5+6)$)

Pip Selection Method 2

Press **CTRL+ALT+SHIFT+3KEY** and every triangular face will be selected. (Every pip + The 8 corners of the dice. Just deselect the 8 corners and you're good to go!) ;-). Similarly, you can select quad faces using the **4KEY**.

Pip Selection Method 3

You can also use circle select to easily select the desired faces. Enter circle select mode by hitting **BKEY** twice. The circle can be made bigger or smaller by using the **scroll wheel**. Drop out of the mode (**RMB**) to rotate the cube and drop back in as above. Selected vertices are added to those already selected like with lasso mode but without the need to keep holding the control key or draw an accurate lasso)



Figure 317

Turn subsurf back on (*Modifiers panel*→subsurf modifier→enable in edit mode) and render **F12** with OSA (only put it up as high as you need for the resolution of the image you are rendering).

(Note: in Blender 2.44 you should use the Assign button in the editing panel (Link and Materials))

(another noob says: don't be fooled (as I was!) by the fact that there are TWO Assign buttons in this tab, you want the big one on the right under materials, not the small one on the left under Vertex groups!!)

Pip Selection Method 4

That's the easiest way: 1- select one of the triangles (the face) 2- **Shift+Gkey**, then choose perimeter or press **4NUM**.

48.6.3. Extra

The reason I modelled the die this way is because it is also very easy to change the sizes of the components e.g. the bevel and the pip size. You do this by selecting the vertical or horizontal segments and just scaling them in one axis. Here we will reduce the pip size and the bevel by half.

Go into front view (**NUM1**), turn off clipping (i.e., allow selection of invisible vertices) and select a line containing pips (i.e., Select vertices mode (**CTRL+TAB+NUM1**), and box select a thin vertical line of vertices to the left of a row of pips, then box select another thin vertical line of vertices to the right of that row of pips, making a total of 64 vertices). Then just scale in one axis e.g. **SKEY**, **XKEY**, 0.5. Remember to have your pivot point set to median:

Do this horizontally and vertically around the die. You should need to scale 9 times for the pips and 6 times for the bevel:

(Noob note: I find this confusing. what is a line containing pips? does this mean a loop line? Using alt-RMB no longer works to select a loop, but selects a single edge.)

(Noob response: he meant a pair of loops of vertices adjacent to a row containing pips. alt-RMB no longer seems to select complete loops - it works until it hits a pip and then it stops - but the box selection can be used.)

(Another Response: Any time you change the geometry of a shape you effect how the automated tools will work. Many dont work at all once you get to complex organic shapes, so its best to not rely on them too heavily.)

You may need to add extra geometry once you are satisfied with the sizes of the dots and the bevel so that the edges of the die don't look warped due to the subdivision. You can use face loop cut again for that and add extra lines in the middle of the gap segments.

49. Model a Goblet

Why a goblet? After all, goblets (fancy containers for consumable liquids) are not really all that important in the real world. However, they represent a very large class of interesting objects: those that are radially symmetric. Also, they are fun to play with.

To model a goblet and actually see the result, we do the following steps:

- create the actual mesh of the object.
- apply a *material* to the object.
- create and light a scene to display the goblet
- render the scene.

The following tutorials demonstrate three different approaches to creating the mesh model, and then show two different materials (glass and silver) that may be applied to the goblet. Finally, we add the minimal elements to the scene to permit an "interesting" rendering.

49.1. Three ways to build the mesh

As a Noob, you may wonder why we describe three different ways to build the mesh. After all, surely the Pros know of a "best" method?! The answer is that there is not really a best method. Blender provides a sophisticated toolkit, and different blender artists will become familiar with different tools. If, after you gain experience, you become comfortable with cube extrusion, then this may become your preferred tool. If you need more than four vertices per "circle" you may find that cylinder extrusion is better for some particular object. If you are comfortable with "spinning" an object, then spinning may be right for you. All three techniques generate an object that is defined by a set of vertices on a set of circles that are "stacked" on a common axis.

After we build the mesh using one of the three methods, we can apply any number of fancy techniques for texture and rendering. We provide two simple "cookbook" approaches in this section so you can see the result, but materials are treated much more extensively in other sections of the book.

In the next three sections of the book we describe the "cube extrusion" approach, the "cylinder" approach, and the "spin" approach. Following sections describe three textures that can be applied to objects that are created with any of the approaches, and the final tutorial provides a simple way to render the result.

The "cube extrusion" approach is a basic technique that all blender artists will become familiar with, to model most types of objects. Since it is such a basic technique, it may become the most efficient way for you to create objects, and you may choose to use it wherever possible rather than switching to a less familiar technique. You can see from the tutorial that this technique is perfectly acceptable for goblets, and by extension it will work for similar objects.

You may prefer the "cylinder extension" technique when you need more than four vertices per circle, because cylinder extrusion is very similar to cube extrusion. Thus your proficiency with cube extrusion will directly carry over to cylinder extrusion.

"Spinning" is conceptually different, even though it creates the same mesh as cylinder extrusion. Spinning is the easiest way to create a radially symmetric model from a two-dimensional description of the cross-section of an object.

49.2. How many circles?

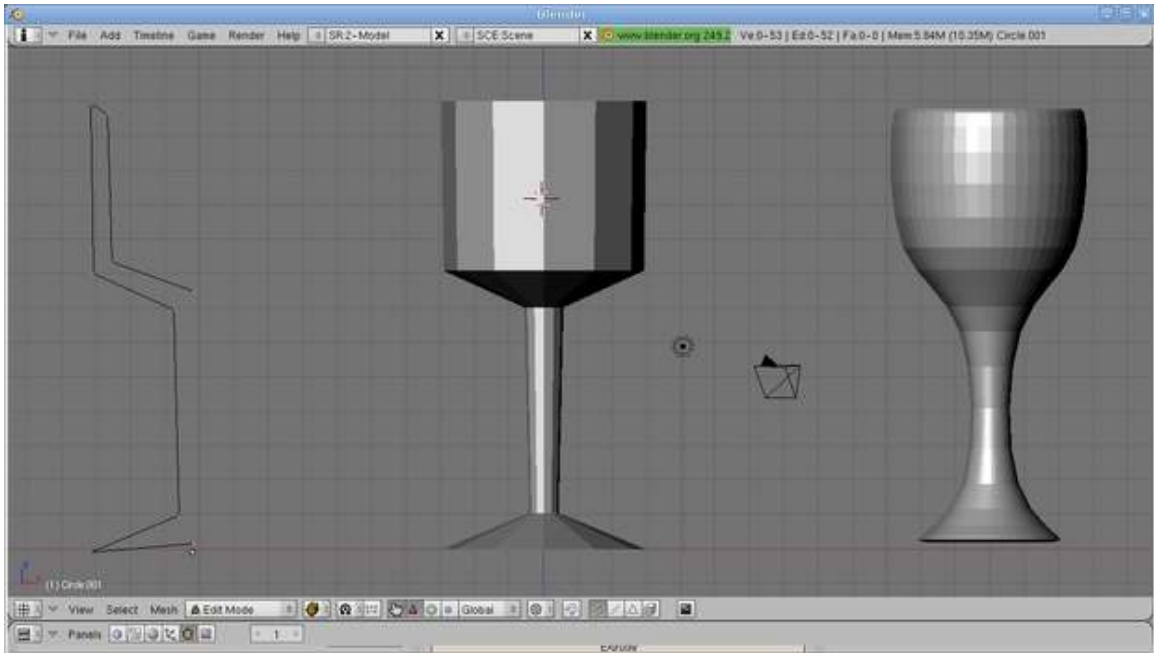


Figure 318

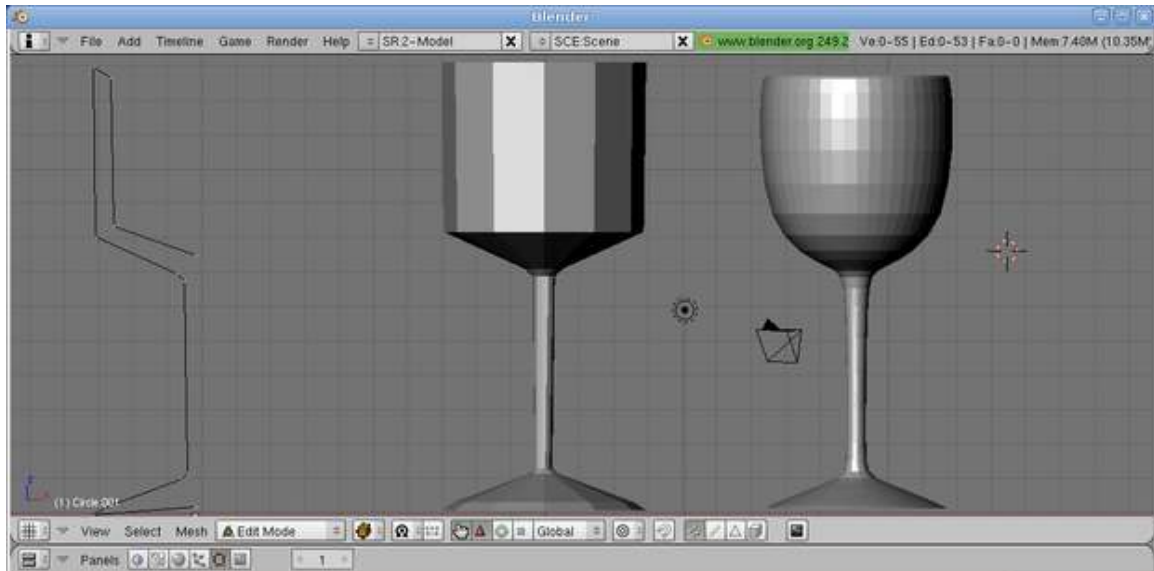


Figure 319

Each of the three methods defines a set of stacked circles, with vertices equidistant around each circle. The "cube extrusion" technique has four vertices on each circle, while the cylinder and spin techniques have a user-specified number of vertices per circle. But how did we pick the number of circles and the spacing between them?

The answer is that we know in advance that we intend to use a technique called "subsurf". Subsurf interpolates additional vertices between those we explicitly specify, and this "smooths out" the profile of our goblet.

This will cause problems where we need a fairly sharp "bend" in the profile, as between the base and the stem and between the stem and the "bulb" of the goblet. At each point in the profile that has a (fairly) sharp bend, we need for the vertical circles to be close to each other. The two images illustrate this effect. Each image shows an outline of the object, the spun object, and the result after subsurf (level 2) has been applied. The outlines and the spun images are nearly identical, but the subsurfed result is very different, because there are three extra vertices in the second outline: one just above the edge of the base, one at the top of the base, and one below the bulb: these are the three locations where we need relatively sharp changes in contour. When we spin the outline, the first outline generates seven circles and two degenerate circles, while the second outline generates ten circles and two degenerate circles. Those extra three circles make all the difference. The same phenomenon occurs with the other two modelling techniques.

50. Model a Silver Goblet

50.1. Basic Shape

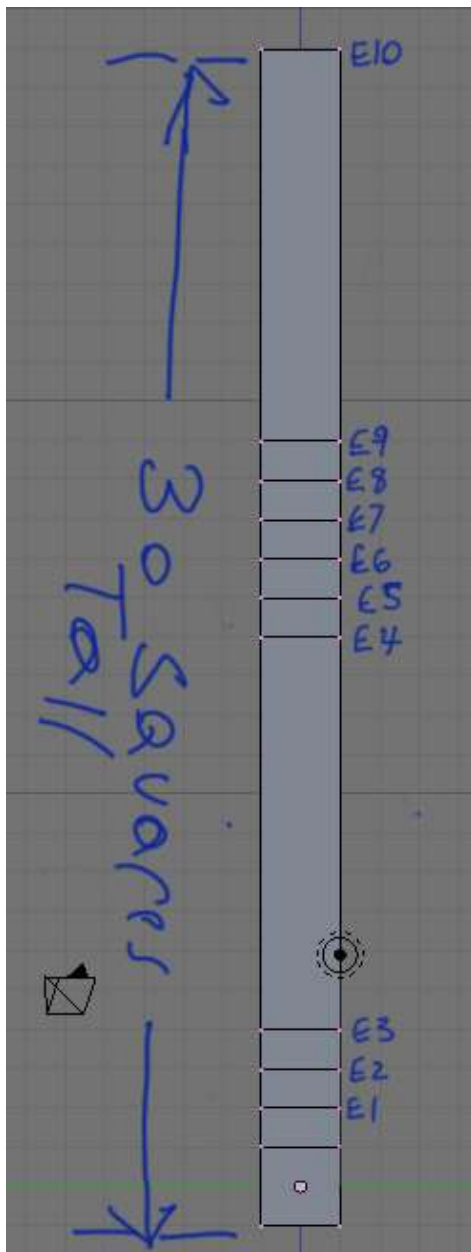


Figure 320

At first glance, the goblet looks like it is composed of cylinders. However, while it is possible to model the goblet with a cylinder mesh, it is easier to make a goblet by using cubes. Cubes make the goblet faster to make and it makes fewer vertices to track. Now, let's start making the basic shape of the goblet.

Start with the default cube and go to Edit Mode (**[TAB]**). Move to the side view (**[NUM3]**). Box select (**[B]**) the top edges of the cube. Extrude (**[E]** → *Regions*) upward about one grid square. Hold down **[CTRL]** while extruding, for incremental movement, and press the **[Z]** to restrict movement to the Z-axis. This extrusion is called E1. Repeat 2 more times for extrusion 2 and 3 (E2 and E3).

Now Extrude a longer piece upwards, for the Goblet's stem, of 10 grid squares (E4). Next we will define the area for the 2 top knobs and the bottom of the glass by Extruding upwards 5 more times at 1 grid square each (E5-E9). Next do another upward Extrusion of about 10 grid squares (E10). This is the actual glass itself.

50.1.1. Inflate the Glass

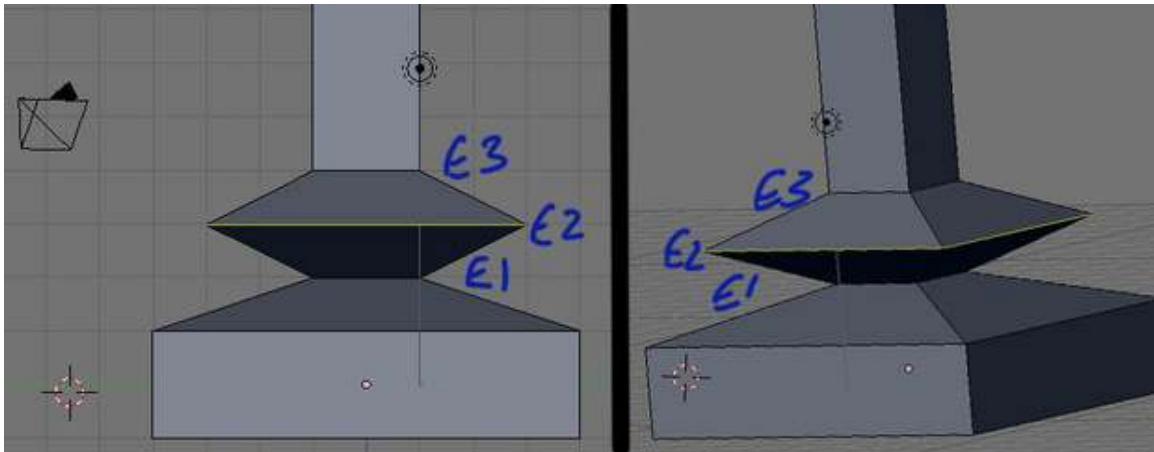
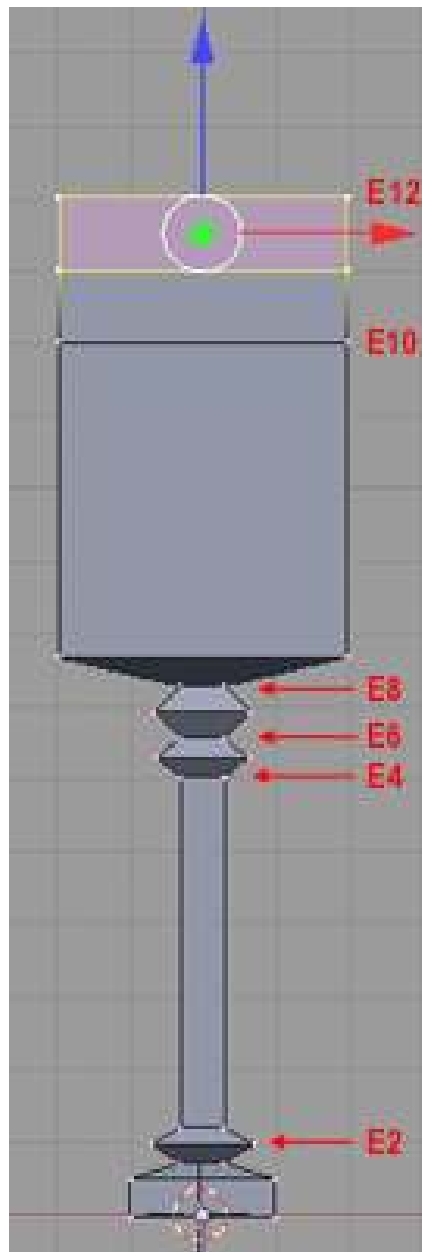


Figure 321 Sizing the first knob

Now, let's begin to inflate the glass. First, clear all selections by hitting the **[A]**. Make sure you are still in the side view (**[NUM3]**). Box select (**[B]**) the bottom most cube (all 8 vertices, not just the bottom 4). Then expand it outward by scaling (**[S]**), then pressing **[SHIFT]** + **[Z]** to lock/prevent any scaling in height along the Z-axis and finally pressing **[4]** to quadruple its size. Next, deselect all vertices (**[A]**), select (**[B]**) the 4th pair of vertices (E2), scale them (**[S]**), lock scaling (**[SHIFT]** + **[Z]**) and triple the vertices in size (**[S]** and **[3]**).

Do the same for E5 and E7 (the 7th and 9th pairs of vertices, respectively). Then expand (**[S]** and **[SHIFT]** + **[Z]**) the top, Goblet rectangle (E9 and E10) to 6 times its current size (**[6]**).

50.1.2. Make the cup's interior**Figure 322**

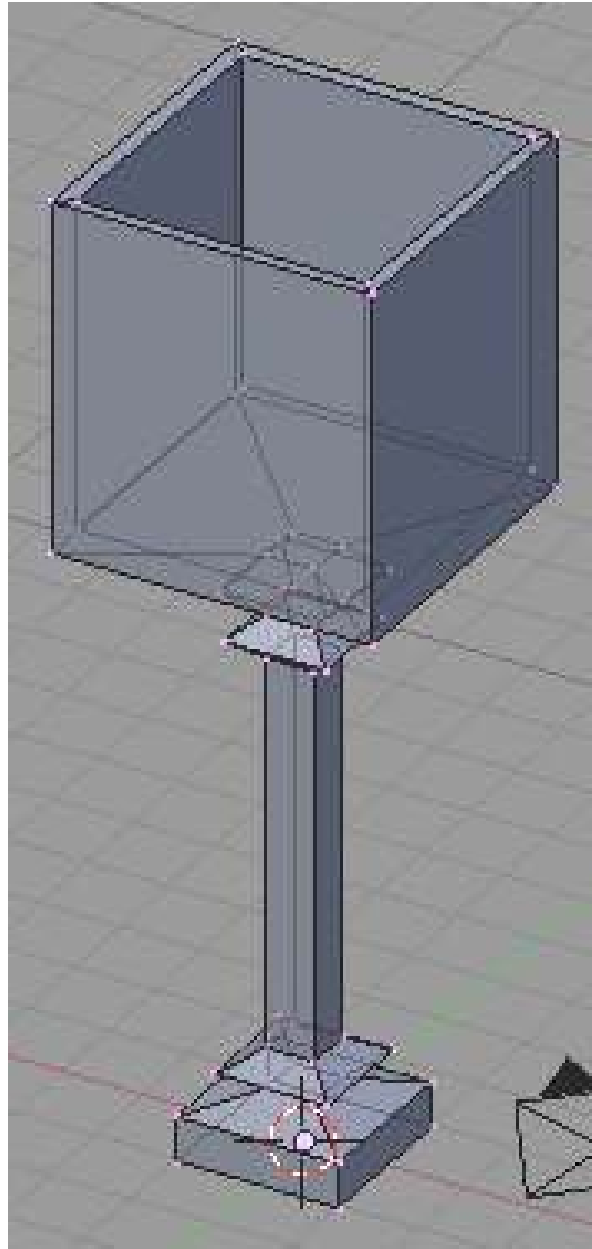


Figure 323

Last is the cup's interior. Once again, you should still be in the side view (**NUM3**). Box select (**B**) the very top most vertices of the Goblet (E10). Once selected, get a better view by changing to 'Orbit Up' (**NUM8**). With the top surface selected, initiate an Extrusion (**E** → *Regions*), followed by a termination with the **ESCAPE**. It will appear as if nothing has happened, but new, overlapping edges have been created and are now selected (this also creates E11). Next Scale (**S**) the selected vertices to 4/5ths of the original size (**.8**). This creates the inside lip of the cup. Now, Extrude (**E**) the interior lip, along the Z-axis (**Z**) downward (**-10**) to create the bottom interior of the Goblet (E12). Finish off with a 'Remove Doubles' (**W** → **Remove Doubles**), for good measure. It should look something like this.

[Noob note: For some reason, when I extruded the interior of the cup to -10, I would have a lip sticking out of the bottom of the cup when I applied the subsurf. To resolve this, I extruded to -9. I was also left with a space at the bottom of the cup (i.e. you could see all the way to the base of the goblet) so I selected the bottom most vertices, pressed F, extruded and hit ESC, merged, selected the center vertex and translated that down -1. This gave me a nice looking bottom for the cup.]

[**Another Noob Note:** When you create the interior of the cup with .8, be sure to hit **[SHIFT] + [Z]** to restrict it to just the X and Y axis, or when you extrude your glass downwards by 10, it will go through the bottom of the glass]

[Noob Question: Does he really mean extrude the interior lip down along the Z-axis? It would seem to me that you should grab it and drag it down (or (**[G]** then **[-10] [ENTER]**)] [Answer: Yes. Extrude creates another set of vertices. Dragging down the existing vertices won't give the lip and the interior the right shape. Try it both ways and see.]

[Pro Note: If you wish for a more detailed cup, extrude the bottom of the cup, then scale it by .2097. Then translate (in layman's terms, that's "grab") it .7181 along the negative Z axis (or -.7181 along the Z axis). Through the use of creasing and control loop-cuts, you can get a *very* nice goblet.]

Note: If the thickness of the vessel sides or bottom is too thin, the Goblets interior faces may 'bleed through', affecting the exterior of the Goblet when you apply a subsurf.

You may want to try modeling the inside/bottom of the Goblet (sim. to "Die Another Way" tutorial). Turning on "Occlude Background Geometry" (**[Z]**) will be helpful, as it turns off the ability to see & select through the model. The button icon is that of a "ISO Cube" located with the 3 select mode buttons (4 dots, 2 lines, & a triangle) of the 3D Toolbar, in Edit Mode.

50.2. Smoothing and Defining

Time to take the mesh and turn it into a proper goblet. Add a subsurf level 2 (**[CTRL] + [2]**) to the mesh. Change to Object Mode (**[TAB]**) and select the Set Smooth button under *Links and Materials*. The cube-looking mesh will now look like an object that was created from a cylinder. This has removed all our crisp edges, and our goblet is looking very unstable! Let's rectify things by flattening the bottom.

Noob Note: *If your goblet has a bulge in it after applying the subsurf, go to **Edit mode** (**[TAB]**), select all (**[A]**), press **[W]**, and select "Remove doubles". (If a menu for boolean operations appears when you hit the **[W]**, then you're not in **Edit mode**; change immediately to Edit mode and try the **[W]** again.)*

Select the four edges that surround the small circle at the very bottom (the very lowest set of edges) and press **[SHIFT] + [E]** to enable creasing. Now drag the mouse up and down to select the level of the crease. When you're satisfied, hit **[LMB]**. Repeat the process for other edges you want to be sharp. I've turned on Draw Creases under *Mesh Tools 1* to illustrate which edges have been creased (highlighted yellow) in this example.

Note: Due to the vagueness of this tutorial I had problems trying to follow what was wanted. I got around it by beveling the goblet (for the cup-edge/thickness), and then merging necessary vertex. Here is how it turned-out:

Goblet¹ Render²



Figure 324 Applied subsurf modifier

1 <http://i88.photobucket.com/albums/k185/captheathmalc/gobletinblender.jpg>
2 <http://i88.photobucket.com/albums/k185/captheathmalc/goblet.jpg>

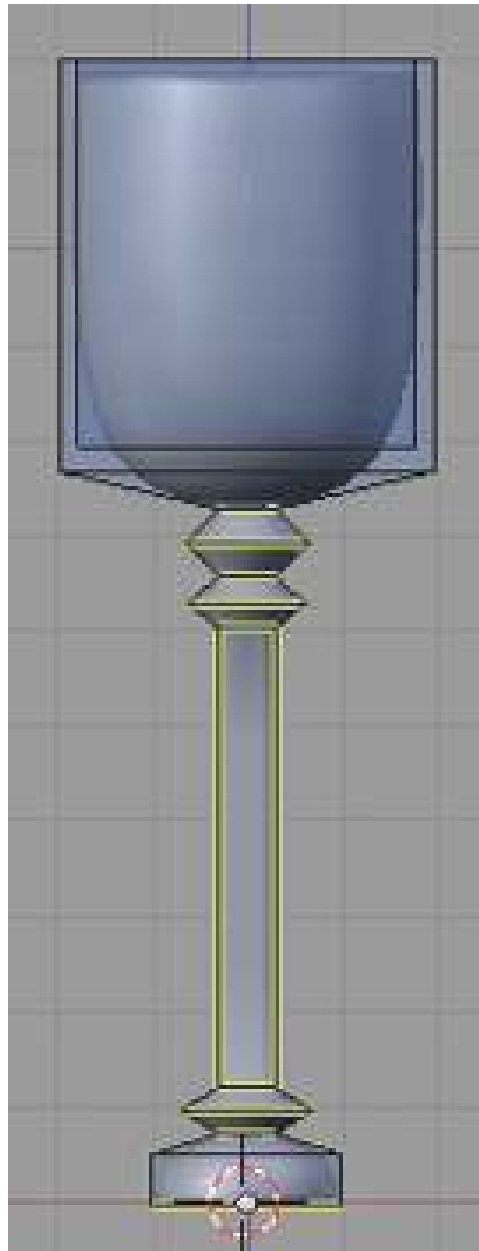


Figure 325 Crease edges

[Noob Note: I found that if you select the bottom 4 vertices and translate down by 2 .5 increments (**Z** -.5 twice) and then you select the bottom 8 vertices (the bottom .5 section) and scale by 1.5 (**S** 1.5) it makes for a more realistic looking goblet base.]

That concludes the creation of the goblet. Save the scene for use in the lighting tutorial. To jump to the relevant lighting section, go to [Blender 3D: Noob to Pro/Light a Silver Goblet³](#)

50.3. Transparency

Okay, this is pretty much just because I wanted to show off my work, but also because transparency was really hard to find and really easy to actually **do**. I don't know if there's a lesson on transparency in this, but I really wanted glass, and I'm impatient. This little bit is for anyone else who's interested in modeling the goblet as glass and is impatient like me. If you try this, first keep the non-transparent goblet saved somewhere, it is used in the "silver goblet and lighting" follow-up tutorial linked above.

Hit F5 to go to the shading context and then click the red ball icon to open the material buttons.

The following images show the settings needed, and the final render of my green glass goblet.

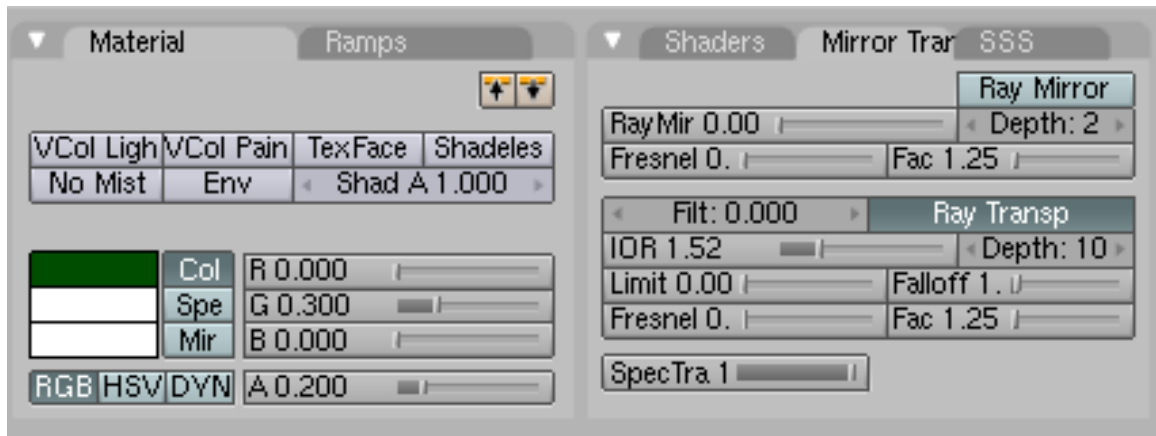


Figure 326

In Materials is an Alpha slider- goes from 0.00 (invisible) to 1.00 (opaque). It needs to be pretty low, less than .25, I'd say. Also in the Edit buttons (F5) is a section called "Mirror Transp", for me it's a tab between "Shaders" and "SSS". In here the "Ray Transp" button must be on. The IOR value is refraction rate- the value of glass is 1.51714. Next to that is Depth, determines how deep to apply transparency. I recommend maxing it to 10, to play it safe.

- Tilescreen's addition

Ray transp stands for Ray Tracing Transparency, this basically involves shooting light out from the camera and bouncing it about your 3d scene. The depth value of the ray tracing settings determines how many times a ray of light can 'mirror' (reflect off a surface) or 'transp' (refract through a surface, just like at school with prisms and the light bends). It is possible to work out how much depth you need as each time it enters or exits a surface you will need +1 level of depth. So for this you should need about 4 or 5 depending on how complex you have made this goblet. If you don't have a sufficient depth it will just come out a solid colour with no transparency.

More on Ray tracing. Samples refers to how many beams of light will be fired out of the camera to calculate the refraction and reflection. higher numbers = better quality but also increase render time, but the real killer is the depth. Increasing it by one means it has to work out an (in my case) an additional 18 bounces. Also try adding a filter to slightly blur your reflections and refractions but only a tiny one.

To get a softer shadow then click on the lamp, press F5 and increase the samples (found in Shadow and Spot) this will shoot out more shadow rays and give you softer shadows.

- End of my addition
- Yoshi's addition:

Try experimenting with the colours. The easiest way I know to do this is:

Select the goblet in object mode. Go to the textures tab (next to the red ball), and under 'texture type' pick 'blend'. Then click 'colours' next to the textures subwindow, and hit 'colorband'. You can change the colours by click on the left of the bar, then double clicking the coloured bar under it. Once you pick the darkest colour (left), click the far right of the checkered bar and do the same as before.

- End of Yoshi's addition



Figure 327

If you want a shadow, you'll need to create a plane and go into the "Shader" settings for it. Turn on "TraShado", otherwise the shadow will be a flat black one and look all wrong.

Noob says: Even when I turn on TraShado, the shadow is flat black... I can't get those lighter edge shadows. Solution: Make sure that TraShado is applied to the plane rather than the cube and that Shadow is also selected.

Noob Note: Do not try and use this goblet for a fluid simulation it will not work.

Reader note: If the ray tracing transparency does not work for you, make sure that the ray tracing is enabled in the *Scene* () menu. It's the *Ray* button in the *Render* panel.

Noob says: No matter what I do, my goblet stays opaque and my goblet has no shadow when on a plane. Do I have to somehow link my goblet to my plane?

Noob answer: Adjust the alpha () and look in the material section. Adjust the (A) slider under RGB sliders (bottom right) to something like 0.25.

Noob Answer 2: Try changing the “Filter” value under the Ray Transp button from its default value of zero to something like 0.1 or 0.2. This will cause the light to take on more of the colour of the object each time it passes through. It should give you a nice, deep colour for the glass, and its shadow, to the point where the alpha value makes no further difference, and can be set to 0.

Here's the final result:



Figure 328

Noob Note: If you have problems with a sandy texture showing up (like in the picture above) your cup is clipping through the plane (bottom face is trying to occupy exactly the same space as the plane), just move it up a little bit on the Z axis, and it should go away.

51. Model a Silver Goblet cylinder

51.1. Preliminaries

This is a version of "Model a Silver Goblet" but starting from a cylinder rather than a cube.

Start Blender or start a new scene (**CTRL-XKEY**) and delete the default cube (in object mode, press **XKEY**).

Change to Top view (**NUM7**), make sure you're in Object mode and add a cylinder (**SPACE** Add > Mesh > Cylinder): in the popup menu, change the number of vertices from 32 to 16, the Radius from 1.000 to 1.800, the Depth from 2.000 to .100 and leave "Cap Ends" as it is.

Change to Front view (**NUM1**).

51.2. Description of the modeling steps

The steps in this tutorial are almost all made up of "extrude" and "scale": so, to avoid repeating key sequences every time,

extrude

means

1. make sure "Limit selection to visible" is off



Figure 329 Shows "Limit selection to visible" icon

2. box select the top vertices of the cylinder: press **BKEY**, click and drag **LMB** to make a rectangle around the top vertices
3. press **EKEY**, select "Region" from the popup menu, press **ZKEY** and type in the amount of the extrusion and press **ENTER**; you can move the mouse instead but it is quicker and easier to type it in.

Newbie question: Why select "Region" and not "Individual faces"? If you use "Region", the common edge between two neighboring faces will remain a single edge after extrusion. If you use

"Individual faces", the edge will become two edges sitting at the same location. This may create problems (black lines when rendering).

For example, the following keystroke sequence extrudes by 1.5:

BKEY, click and drag **LMB** to make a rectangle around the top vertices; **EKEY**, select "Region", **ZKEY**, **1KEY**, **.KEY**, **5KEY**, **ENTER**.

scale

scaling is by default restricted to the X-Y plane; although the numbers in the bottom left corner of the 3Dview show Z changing, in fact only X and Y change, and by equal amounts.

1. press **SKEY**, type in the value and press **ENTER** — you can use the mouse instead but it is quicker and easier to type in the number.

51.3. Creating the Goblet

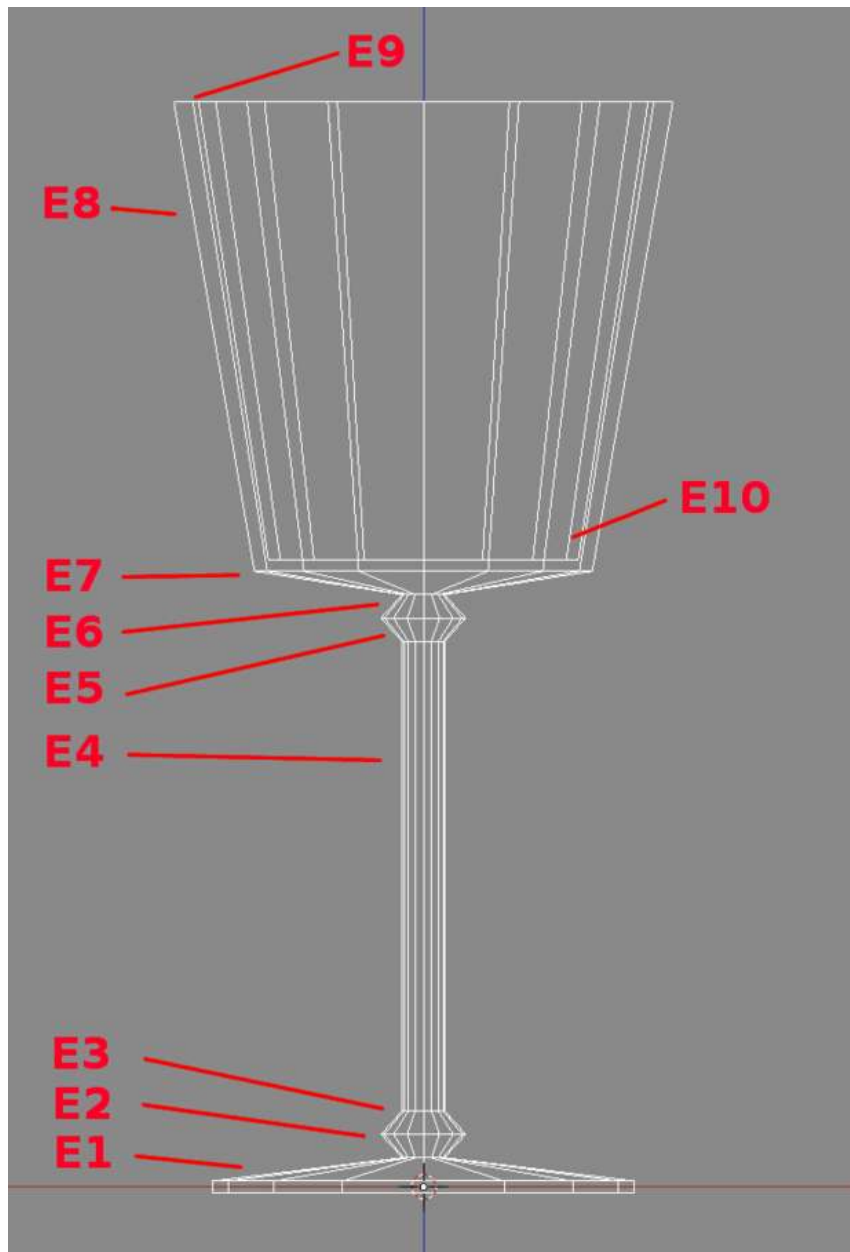


Figure 330 E-numbers and construction steps

This diagram shows the connection between the E-numbers and the goblet construction.

- **E1:** Deselect all vertices (**AKEY**) and extrude by 0.2: you may need to zoom in (**SCROLL-MMB**) to do this as it's quite thin. Scale to 0.1.
- **E2:** Extrude by 0.2, scale by 2.
- **E3:** Extrude by 0.2, scale by 0.5 to make the lower knob.
- **E4:** Extrude by 4 to make the stem.
- **E5:** Extrude by 0.2, scale by 2.

- **E6:** Extrude by 0.2, scale by 0.5 to make the upper knob.
- **E7:** Extrude by 0.2, scale by 8 to make the base of the cup.
- **E8:** Extrude by 4, if you wish to make a flared cup, you can scale by 1.5.
- **E9:** Extrude by 0.0, scale by 0.9 to make the rim of the cup. (This will create a new ring of vertices and then move them in towards the centre.)

Now go into Wireframe mode (**ZKEY**) so you can see inside to guide the next few steps.

- **E10:** Extrude by -3.9, that is, downwards. and scale by 0.69: you can do this last scaling with the mouse, if you like, to get the edges of the inside of the cup and the outside parallel.
- **E11:** Extrude by 0.0, scale by 0.0 to make the inside of the cup.

You now have a goblet, the base of the inside of the cup is the face of the last extrusion, is circular and flat as it derives from a cylinder.

If you haven't already saved your work-in-progress, now would be a good time.

51.4. Subsurfing and smoothing the goblet

The last step is to subsurf and smooth: go into Object mode (**TAB**) and enable Solid mode again (**ZKEY**).

Select the Editing panel from the Buttons window (**F9**) and, in the "Modifiers" panel, click on "Add Modifier", select "Subsurf" from the popup menu and, in the Subsurf display, increase "Levels" from 1 to 2.



Figure 331 Shows the Editing panel icon and the Modifier panel

At the bottom right of the "Links and Materials" panel, click on the "Set Smooth" button. At the bottom of the cup you will see fluting — this is an artifact caused by smoothing and subsurfing triangles on a curved surface. Here it adds to the appearance, don't you think?

Noob Note: If your cup does not look smooth after using Subsurf and Set smooth like mine, you may have to delete all but two of center vertices (bottom vertex at base of cup and vertex at the bottom inside the cup - **AKEY** to select all vertices, **WKEY** and Remove Doubles).

51.5. Flattening the base of the goblet

The base of the goblet is curved due to the subsurfing, so needs to be flattened.

Go into Edit mode, deselect all (**AKEY**), box-select the lowest set of vertices (**BKEY**), then crease (remove the subsurfing) by pressing **SHIFT-EKEY** and typing "1" followed by **ENTER**.

The final result should look something like this:



Figure 332

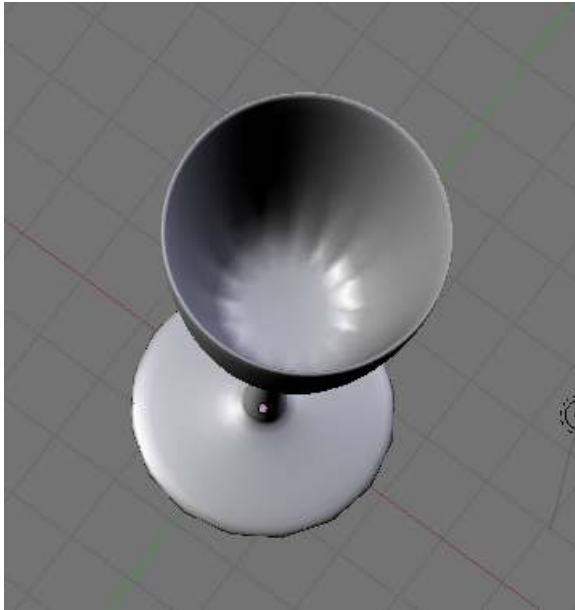


Figure 333

Save the scene for use in the lighting tutorial. To jump to the relevant lighting section, go to Blender 3D: Noob to Pro/Light a Silver Goblet¹

¹ Chapter 53 on page 397

52. Spin a goblet

The **spin** technique is a good choice when you want to model an object that is radially symmetric and you know what the cross section of the object looks like. This is the virtual equivalent of using a lathe¹ to create an object in the real world. With the spin technique, we draw an outline of one half of the outline of the object, and then spin the outline about an axis to create the object's mesh.

¹ <http://en.wikipedia.org/wiki/lathe>

52.1. Modeling



Figure 334 Photo of a real-world goblet

Here is a real-world goblet. The picture is not an orthographic image, so we cannot directly copy the outline, but it is close.

52.1.1. Setup

We now model the outline. Let's start with the default model.² We want to create an initial object consisting of a two-dimensional outline, so select the cube, go to edit mode, and delete all the vertices. This leaves us with an object that has no vertices as a nice place to start. We want the resulting goblet to sit at the origin with the Z axis as its axis of symmetry so we get a front view.³ Since we will be working directly with mesh vertices, the *manipulator* is a hindrance, so turn it off.⁴ (With the manipulator on, it is easy to accidentally move a vertex out of the editing plane, but we want a 2D cross-section.) We are now ready to create the two-dimensional outline as a chain of vertices with the first and last vertices on the axis.

52.1.2. First Vertex

Now, place the first vertex: this will be the center of the bottom of the goblet, so place it slightly above the origin on the Z axis.⁵ Why? Well, for two reasons: in the real world, the bottom of a goblet is not actually flat. Instead, the rim is lower than the center of the bottom, so the bottom is concave and the goblet sits on a flat surface without wobbling. The second reason is that we intend to use the "subsurf" technique, and this technique will make a flat surface slightly convex in our virtual world, so we will preemptively start the bottom of the base with a slightly concave surface.

52.1.3. Finish the outline

Add the second vertex, which will be on the rim of the goblet's base: move the cursor to the X axis at about -3, and add a vertex.⁶ Add additional vertices to your outline by moving the cursor to the desired location and adding,⁷ for as many vertices as you need to accurately model the outline. Since you will be using subsurf later, make sure that you place two vertices near each other when you need a sharp curve in your outline. Otherwise, subsurf will convert your sharp curve into a wide smooth curve.

Noob Note: This is wonderful advice, but it won't work, because Blender interprets the **CTRL** + **LMB** differently when you attempt to place an extruded vertex too near the vertex from which you are extruding. One quick way around this is to create your vertices sufficiently far from each other to avoid the problem, and then come back and move them⁸ to the correct spots later.

Eventually, you will place the last vertex, which will become the point at the bottom of the inside of the goblet. This point should be on the Z axis. Your outline is done!

2 Default model:select *file* → *new*.

3 Front view:press **NUM1**

4 manipulator off: press **CTRL** + **SPACE** to bring up the *Manipulator* menu, then select *Enable/Disable*

5 Place a vertex: **CTRL** + **LMB**.

6 Add a vertex:press **CTRL** + **LMB**, or **E**, **ENTER**

7 Add a vertex:press **CTRL** + **LMB**, or **E**, **ENTER**

8 Move a vertex: select it with **RMB**, move it with the cursor, and then press **LMB**.

52.1.4. Spin

Now, to spin it. first, make sure that you are still in edit mode, and that the last vertex is selected. Set the cursor to the selection.⁹ Now, get a top view:¹⁰ you are now looking at your outline from the top, and it should look like a straight line along the X axis with one end at the cursor on the Z axis. (If this is not the case, select and move points to the X axis and check your work by switching back to the front view, then come back to the top view.) Now select the whole outline¹¹ and then move to the button menu to perform the spin. Set the rotation to 360 and the count to 12 (or another number of your choice) and click on "spin." An elaborate circle will appear. Go to the front view¹² to see your un-smooth goblet.

52.1.5. Finish up

Since you are in edit mode and you have the whole mesh selected, this is a good time to remove duplicate vertices. The two vertices on the Z axis (base bottom center and bulb bottom center) were duplicated 12 times, and the entire outline was duplicated once when the circle closed at the end of the spin. Remove the duplicates.¹³ If you fail to do this, the subsurf operation will create a cusp at the two centers, and a crease at the duplicated outline.

If you placed the two Z-axis vertices by hand, they may not be exactly on the axis, and therefore may still be duplicated, leaving a tiny "hole" at the axis. Fix this by merging each of the two sets separately:

1. select the vertices to merge.
2. merge at center.¹⁴

Now start to smooth by using the subsurf in the button window. First add a modifier and select subsurf. Set levels to 2 Then, use *smooth* in the texture window. You now have a goblet model.

9 Cursor to selection: (**[SHIFT]** + **[S]** , then *cursor*→*selection*

10 top view: **[NUM7]**

11 Select all: **[A]** once or twice, until all are displayed in yellow

12 Front view:press **[NUM1]**

13 Remove duplicate vertices:press **[W]** and select *Remove Doubles*

14 Merge at center:Press **[W]**,*Merge* → *At Center*, confirm with **[LMB]** or **[ENTER]**

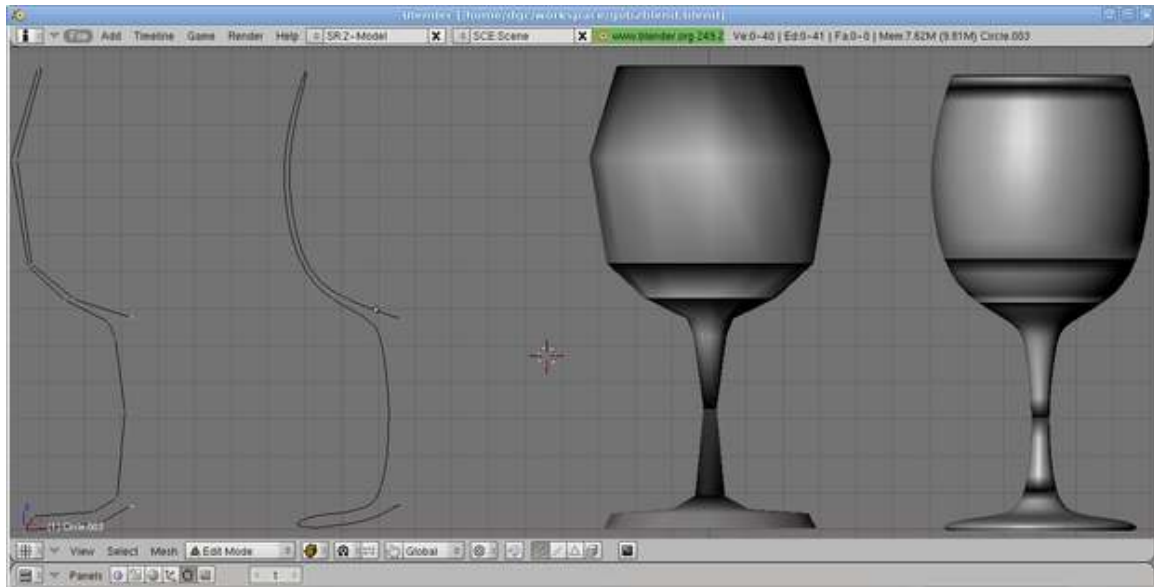


Figure 335

The image shows four objects: the two-dimensional outline, a smoothed version of the two-dimensional outline, the result of spinning the outline, and the result of subsurfing and smoothing. As you follow the procedure above, you will not actually have more than one object at a time as shown here.

52.2. Rendering



Figure 336

As with the other two goblets, It is difficult to fully evaluate the model unless you render it. However, a "pretty" rendering requires at least minimal materials, lighting, and scenery. This section is a cookbook approach to providing these minimal elements and is not really a useful tutorial, so we won't explain the concepts. These topics are treated at length in later tutorials. If you wish to explore these subjects in more depth, go to the appropriate tutorials.

- The Goblet material: plain glass.
- The tablecloth: A (non-existent) table with a white tablecloth, in a featureless room painted yellow.

- The lighting: one bright lamp.

52.2.1. Material

- Go to object mode and select the goblet.
- In the Materials button window, add a material¹⁵ to the goblet, and name it "glass."
- In the *Material* tab, set the color to black (set R, G, and B all to 0.000) Here, "Black" merely means "do not add any color." It does not mean that the goblet looks black.
- Set the Alpha to 0.1 (i.e., quite transparent.)
- In the "*Mirror Transp*" tab, click the Ray Trace button, set the IOR to 1.5, and set Depth to 6 (or higher on a very fast computer.)

52.2.2. Tablecloth

- In the main window, put the cursor at the origin and switch to top view. In object mode, add a plane and then scale it to quite large. This is your tablecloth
- Go to object mode and select the tablecloth.
- In the *Material* tab, set the color to white and set the Alpha to 1.000 (i.e., opaque.)
- In the *Shaders* tab, click on *Trashado*. (**2.61**: *Shadows* tab, *Receive Transparent*.) This allows the tablecloth to show the ray traced shadow of your goblet instead of a fake shadow.

52.2.3. Camera and lighting

Render the scene. If you started with the default camera and light the scene will be too dark and the shadow effect from the lamp will not be too pretty. To fix this:

- move the lamp higher and farther away.
- turn up the intensity.

Now adjust the camera:

- shift to camera view
- dolly and aim the camera
- move the camera back some
- render

52.3. Detailed keystrokes

15 Chapter 36 on page 213

53. Light a Silver Goblet

53.1. Techniques

You should know how to:

- Perform actions discussed previously in the tutorial.

This section will recap or introduce:

- Reflective material
- Positioning camera and light
- Editing the World colors

53.2. Objects in the Scene

Create the goblet discussed in [Model a Silver Goblet](#)¹ or load it if previously made. If you haven't already made the goblet, feel free to try the tutorial using a sphere or something else instead and you will still get a good outcome. In Object Mode with **NUM 7** view, add a plane mesh. Scale the plane to a very large size and make sure the goblet is sitting comfortably on top of it.

Select the camera and move it so that the goblet, its shadow, and its reflection in the plane will be seen. You can see the numerical location of the camera by bringing up the *Transform Properties* window by pressing **NKEY** in the viewport. In my example where 0,0,0 is the bottom center of the goblet, the camera is located at 27, -21, 19 XYZ with a rotation of 63.5, 0.62, 46.7.

Create a Spot with **SPACE** → *Add* → *Lamp* → *Spot* (must be a Spot, as this is the only type of lamp that casts shadows). Move it so that it will create a shadow on the plane. Again, the location of this example is 9, -15.2, 25.8 for XYZ (rotation does not matter for this type of light source).

¹ Chapter 50 on page 373

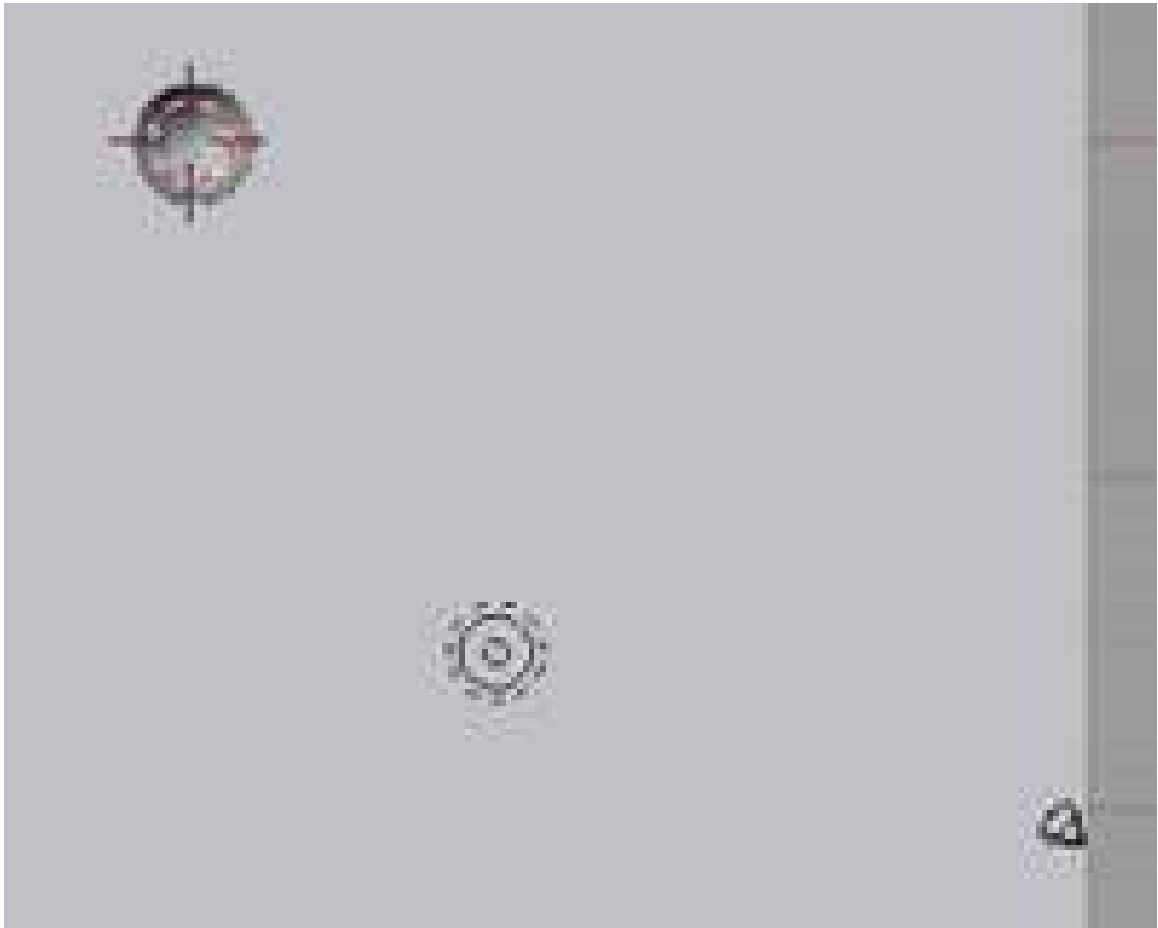


Figure 337 Overhead view of the objects

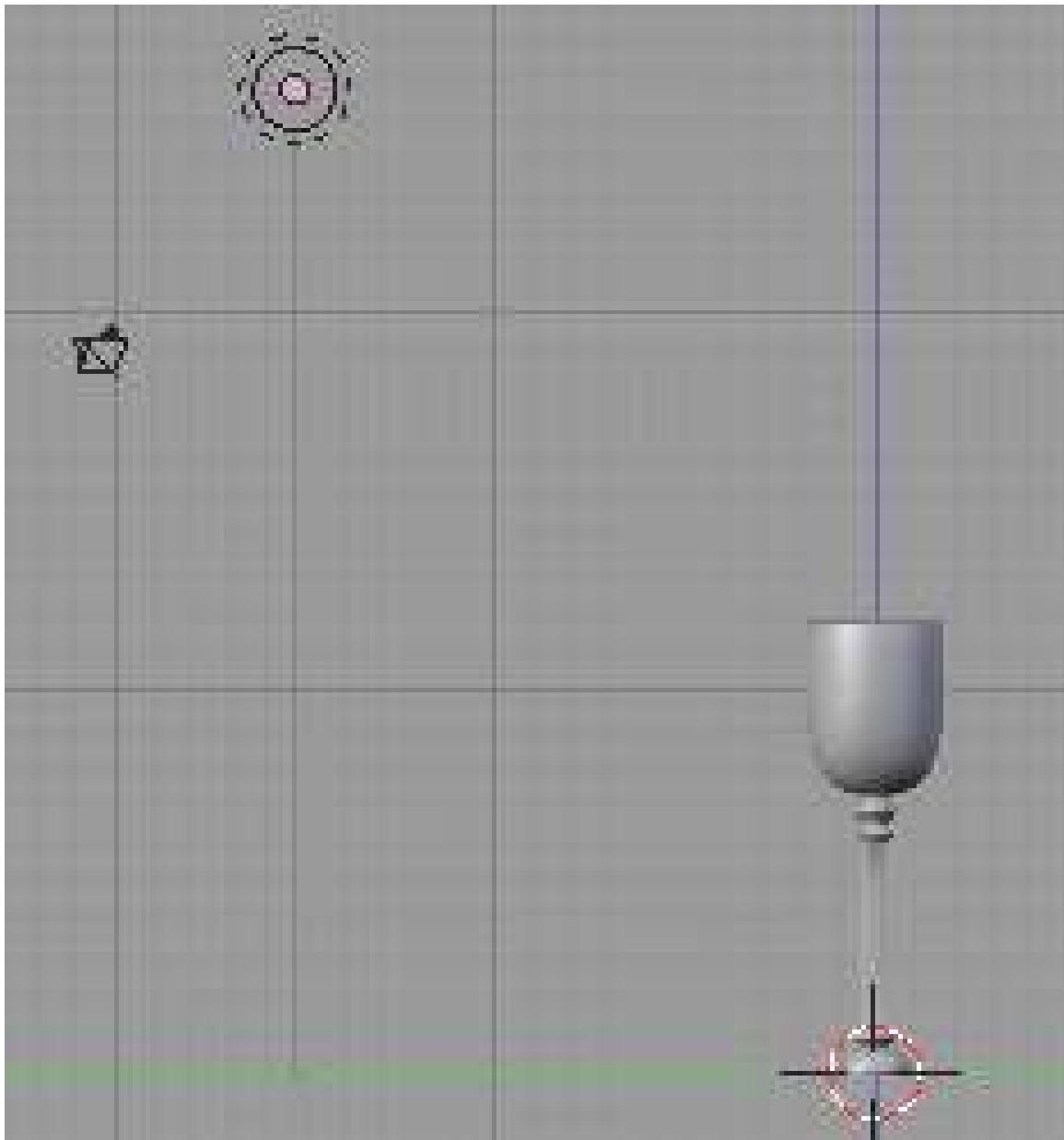


Figure 338 Front view of the objects

The rendering of this scene yields:

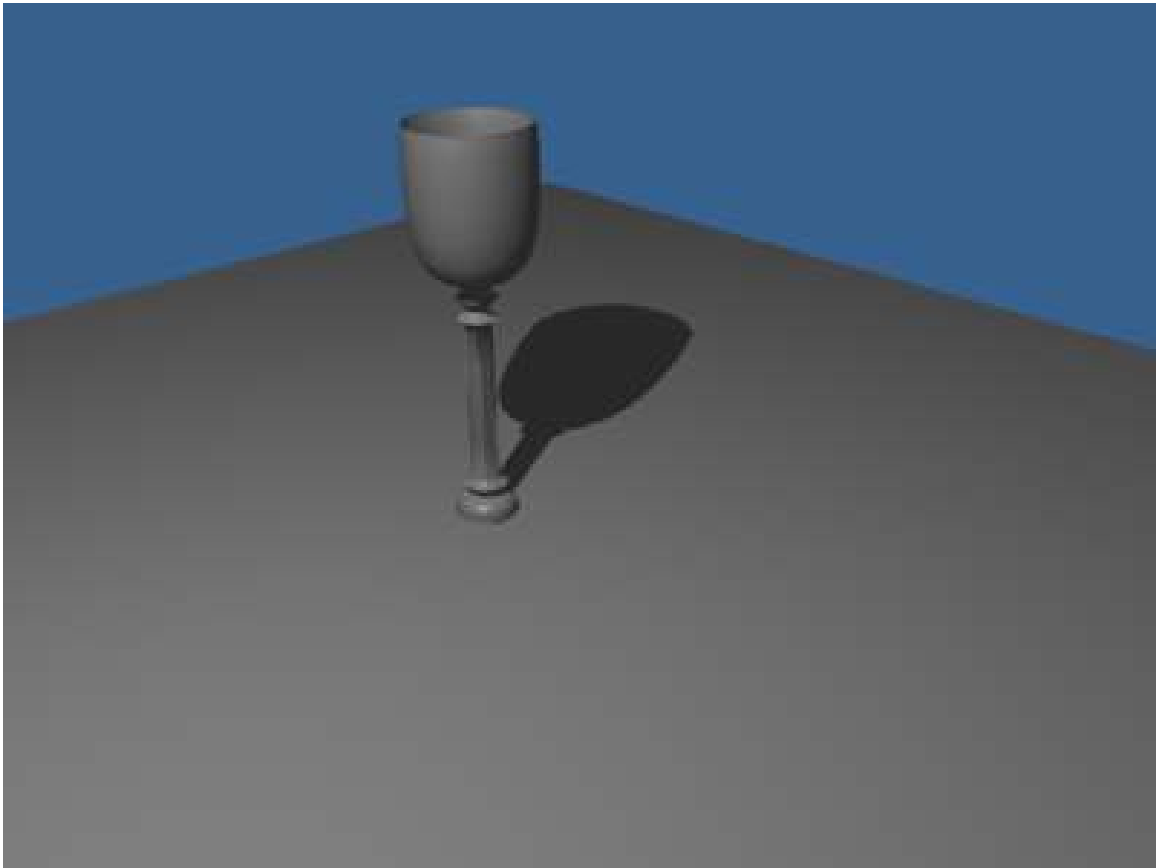


Figure 339 Current render

Note: You must turn on the Ray button in the Render (F10) buttons. Otherwise no shadows will appear!

53.3. Adding the Atmosphere

Noob Note: Many of the options here will be different if you have Yafray selected. In order to change to Blender's internal renderer, go to the Render tab (F10). Under the Render panel, click the drop down menu underneath the Render button. Select Blender Internal from the list.

Noob Note: On my version of Blender (2.49), the yafray button has been removed, so if you don't find it, don't worry.

In Object Mode, select the goblet and go to the Shading panel (F5). If no material is linked to the goblet, add new material. Rename the material 'cup' or something similar. The area of interest is the *Mirror Transp* window. Highlighted below are the mirror options we'll be playing with. Press the Ray Mirror button to make the material act like a mirror and reflect light.

Move the RayMir slide to 0.85 or type it in after **LMB** on the number. This is how reflective the surface will be. A low number of 0.00 means that it reflects little while a high number of 1.00 reflects everything.

Also change the Fresnel slide from 0.0 to 1.4. This will increase the power of the Fresnel function. What this means is the color of the material will be strong because the light source is taken into consideration. If the Fresnel wasn't used, the object would appear dark because the light source isn't directly calculated in the mirror. Also, change the color of the goblet to white. Using a light color will give your goblet an interesting patina if you so choose.

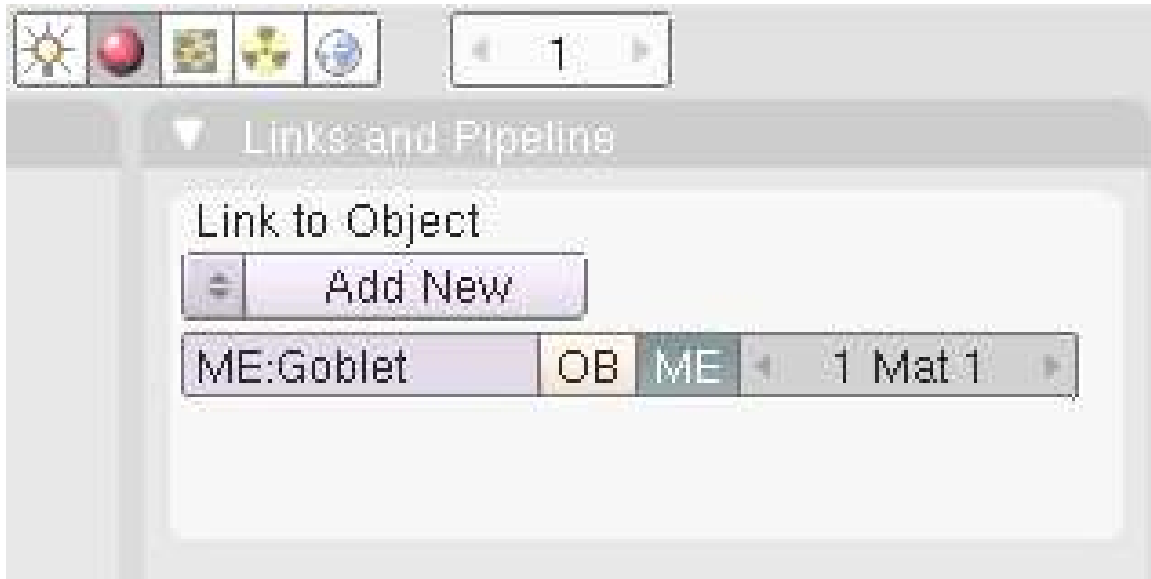


Figure 340 No material linked

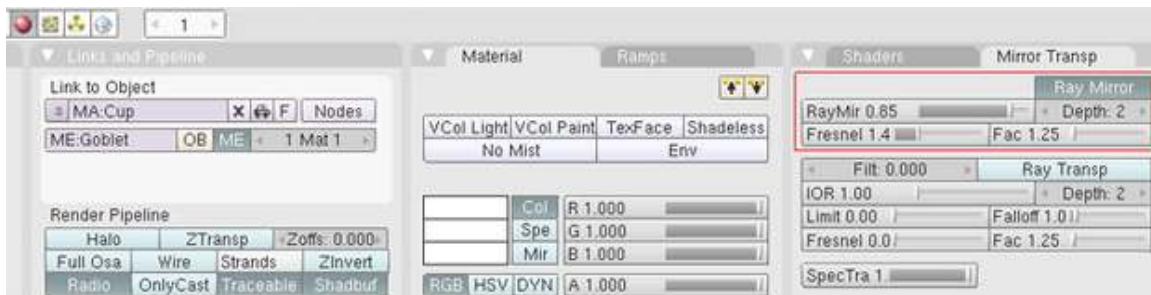


Figure 341 Make the goblet reflective

Next, select the plane and modify the material. We want the plane to be dark and shiny. Set *Col* and *Spe* to near black for the color. For reflectivity, turn on *Ray Mirror* to about 0.15 *RayMir* and ignore *Fresnel* this time.



Figure 342 Edit the materials of the plane

The scene will still be quite dark, so let's turn up the light. Sometimes you want to add additional light sources and other times you just want what you have to be brighter. Here we'll stick to the one lamp.

Select the lamp object and increase the Energy to 3.000 making it 3 times brighter than before. As always, try changing it to several levels to see how bright or dark you can make the rendering.

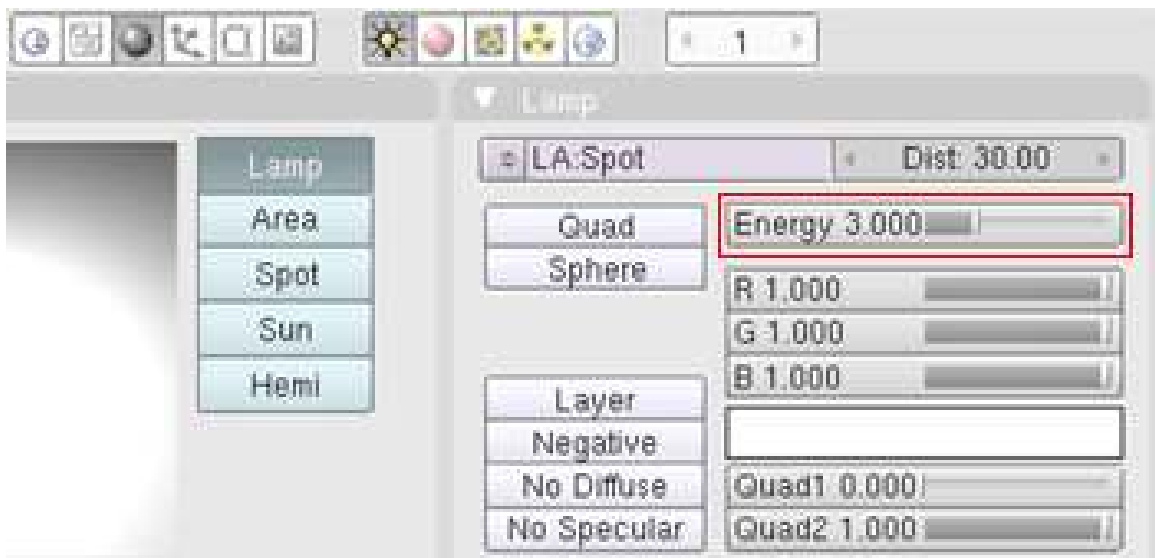


Figure 343 Let it shine

There is only one more thing to do before rendering the scene: change the world. Under the Shading panel there is a globe icon that will get you to the world buttons. Here you have *Real*, *Blend*, and *Paper* buttons under the *Preview* window. And under the *World* window are options for changing the color of the horizon (Ho RGB), zenith (Ze RGB), and ambient (Am RGB). We're interested in these two windows at the moment.

Using *Real* and *Blend* will affect the way the horizon and zenith interact. Experiment with them to see what they do in the preview. In this example, *Real* and *Blend* are turned on.

The *Paper* button works a little differently in that what you see in the preview will essentially be the background of your render. This effect is most noticeable when your camera is rotated. Despite the camera rotation, the preview would still be 'wallpapered' on the render.

For our world, set the RGB close to 0.00 for the horizon, zenith, and ambient.

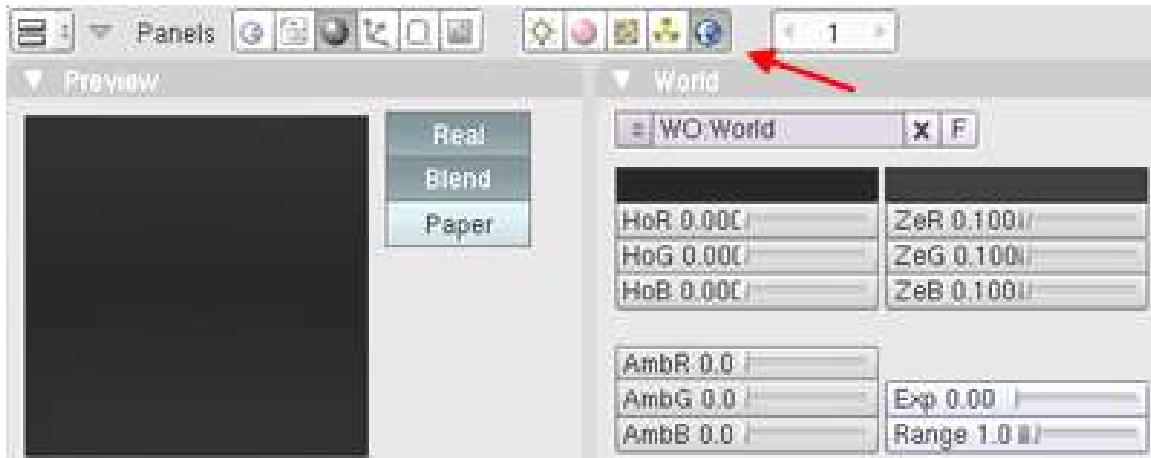


Figure 344 World shading

That was the last step! Make sure the camera is in the right spot and render the scene. Here is the output of this example



Figure 345 Final rendering



Figure 346

Problem: When I render I see brown where I have used "Shift-E" and "G" and I followed everything.

Answer: Please remember, the author did say play with the settings a bit, this tutorial should be used as a guide. We are learning how to use all the tools that Blender has to offer, that is the important thing. It is up to us to experiment more with the settings set forth in these tutorials. If the settings in these tutorials do not give us the same results, that is OK, we should be changing them anyway to express ourselves! Also remember, NEVER strive for PERFECTION, but ALWAYS strive for EXCELLENCE. Perfection only leads to frustration, and it is frustrating enough, at times, to learn something new. Have fun learning, I know I am.

Noob Note: That was touching and true.

Noob Note 2: You just changed my life :D

Noob Question: I managed to get it looking like the first picture above. How do I get it to look like the second?

Pro Answer: Change the reflection settings. a higher depth and a larger raymir value will make the goblet more "mirror" like as in the 2nd picture. the other settings should be left alone, or you can experiment with them to achieve the effect that you want. lighting is also important. the object that is to be reflected has to be illuminated as well as the object that is doing the illuminating. different lights (don't use a hemi if you want it to be realistic) at different angles will give you a more realistic effect. -TMOG

Noob Note 3: On the answer above, I didn't manage to do it with any of the things the pro said in the answer here, I found out that the key is to change the color of the material which is white (or close to white) in the upper picture. In the lower picture the color is set to black (or close to black). This eliminates the "un-metallic" whitish sheen that the goblet in the upper picture has.

note from hash i don't know about those using lower versions but those using v2.5x have a choice of different shaders both for diffuse and specular colours. I strongly suggest that for the same object one should try playing around with the shaders. They can create different effects. for eg. here if you change the diffuse shader type to **oren-nayar** and the specular shader type to **wardiso**, the goblet will have a glossy , finished look.

54. Simple Vehicle



Figure 347 Let's make this jeep.

The idea of this tutorial is to learn to face a complex project. A vehicle is a nice object to use to test yourself and find new problems.

First, we must understand that a project does not reproduce the real world; a project shows an idea or thought and will result in a final image or video. Whatever does not appear in the final result is unnecessary to include in the model.

What vehicle should we make? Let's go with the classic jeep. This will allow for a lot of doodads.

Let's decide what objects of the jeep model will need to be made - body, wheels, seats, and a rocket launcher for good measure. Objects we can ignore include the engine, which remains hidden under the hood. There are many additional objects you can make such as a steering wheel to customize

your jeep.

55. Simple Vehicle: Wheel

Blender3d Aligned to view issue

This tutorial relies on objects being created so that they are aligned to the view that you're looking through. Versions 2.48 and above have changed the way this works. Visit Aligned (http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Aligned_to_view_issue) to view issue to understand the settings that need to be changed.

55.1. Techniques

You should already know how to:

- Make a mesh
- Navigate the viewport
- Extrusion
- Create, edit materials

This section will recap and introduce:

- Forming faces
- Subsurfing
- Merging vertices
- Object naming

For our premise, envision jeep tires. They're not too sleek but rather rugged for all kinds of terrain. We need a tire that can handle any obstacle in its way.

55.2. Model the tire

Hit **NUM1** to set front view (XZ coordinates), then delete the cube.

55.2.1. Create the outside of the tire

Add a cylinder (**SPACE > Add > Mesh > Cylinder**)

use 32 vertices, set the radius to 4, depth to 3, and un-click the "Cap Ends" button.

55.2.2. Create the inside of the tire

Change to **NUM1** view and make sure you are in **Edit Mode**.

With all the vertices selected, hit **EKEY** (for blender 2.5X and on-wards, hit "Alt+EKEY" and chose **INDIVIDUAL FACES**) and on the menu that appears select "**INDIVIDUAL FACES**" to extrude the individual faces into the circle.

[Pro Note: It would be less messy if you extrude the region, then abort the operation with **ESC**, then scale the new faces inward while locking the **Global Y** with **SHIFT-Y**.] [Noob Question: Does doing it the Pro Note's way not create a slightly different shape which stops the ruggedness from working later?]

[Another method by a noob: i think a better way is to select all the vertices and in in edit mode, hit "Alt+E" and select "Edges Only" and then hit "Esc". immediately after that, hit "S" to scale and lock the y axis by hitting "Shift+Y", and then scale inward. after that, the inner vertices should be selected, hit "F" to create faces.] [Another method by a noob: As I was unable to reach the result in this step (I only managed to obtain the tyre INWARDS), I did this way with a nice and similar result: Select the cylinder with A in **NUM1** view, then **Extrude Individual** (press button in Mesh tool) by 0.1 or 0.2 outwards. Go in face-select mode and select all the faces in the inner part of the wire (or select one then **CTRL-G** and select "Area") and press **EKEY** to extrude inwards and to create the tyre shoulder, then, after subsurf, **SHIFT-E** to crease the edge.]

(Another question. If im locking the global Y, the tire won't be rounded. And shouldn't i have to view the tire on top?)

-**comment** In response to the noob questions above, after playing around with my Blender v2.6 I discovered the easiest way to ensure the tire results in the picture below is this: Select all faces in face select mode, then ensure all the face normals are recalculated so that they are all pointing in the same direction (can be inwards or outwards, as long as they're all the same direction. Some faces might be pointing inwards while others are pointing outwards, this means they need to be recalculated. To check the face normals, check the "Face" box underneath "Normals:" in the "Mesh Display" drop down menu in the Window Tool Bar Widget which you press Nkey to show/hide (see image if needed). Once the face normals are corrected, and you have selected all faces, you can then proceed to press the SPACEBAR, bringing up the search menu (for Blender 2.59+), and type in "Extrude individual faces and move" then follow the directions as given below. You can do the search menu, or you can just click on "Extrude Individual" under "Add:" in the Mesh Tools bar (press Tkey to show/hide). << After working awhile on this Blender 2.61 you want this method.*****

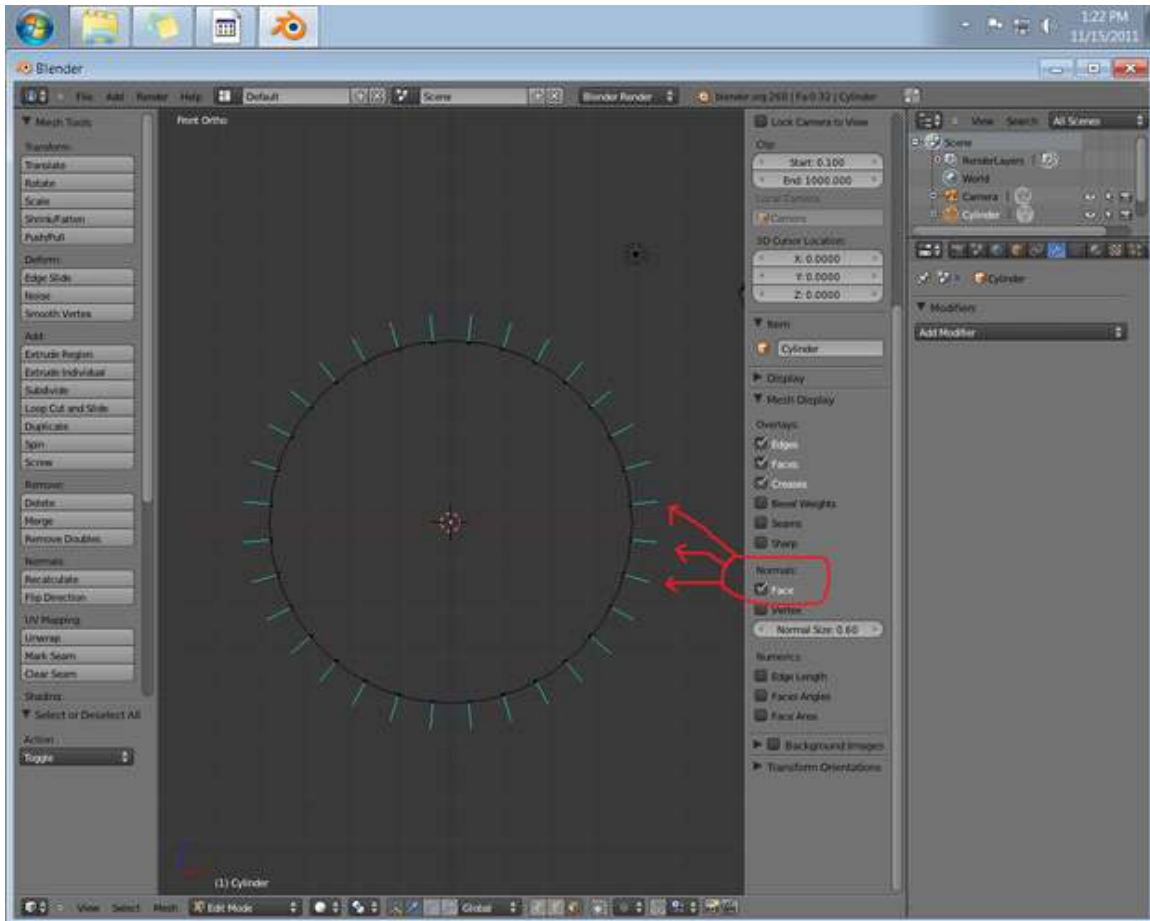


Figure 348 Face Normals in Blender

-**comment** If you do an extrusion/scale without locking global Y, then the walls of the tire will either lean inwards from the outside radius towards the inside radius -OR- lean outwards from the original radius to the now outside radius. In real life, tire walls can either be expanded outwards (as do most 4x4, snow, or street tires), or lean towards the inside of the tire (as do most low-profile tires). If you want this effect, then do not lock global Y; however, if you wanted a perfectly flat wall faces connecting each radii (the outside and inside radius) then you need to lock global Y.

You may either type in -1.2 and hit enter, or Hold **SHIFT CTRL** while extruding the faces until the sides come in -1.200 units (in older versions of blender hit **NKEY** first)

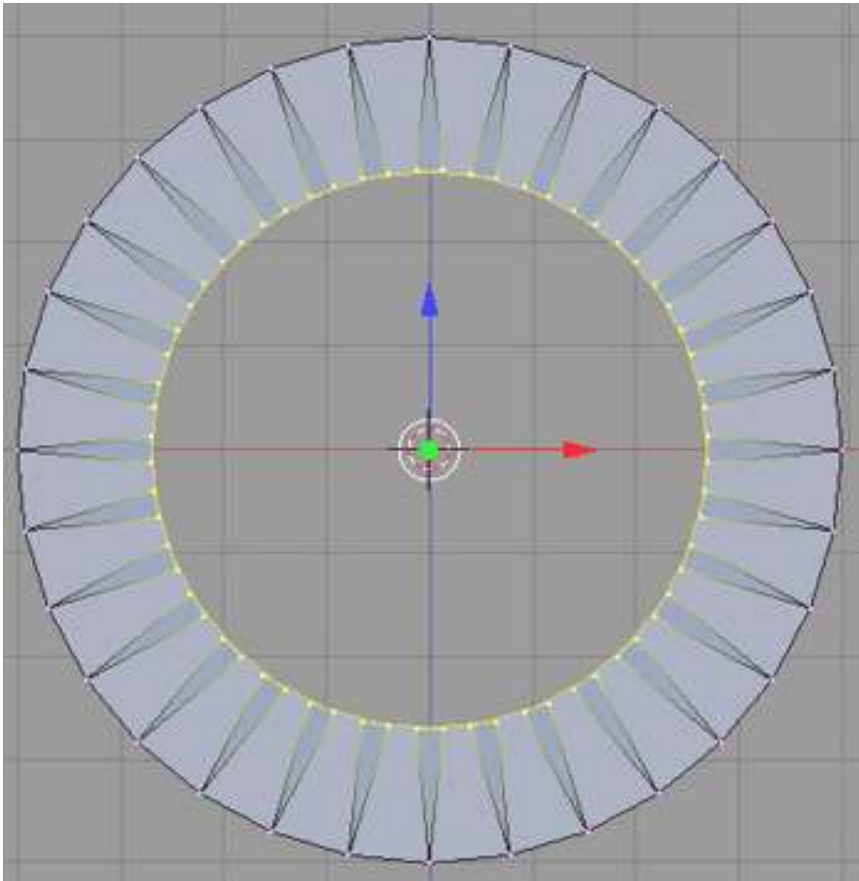


Figure 349 Extruded inwards

55.2.3. Create the outside faces

Rotate around the model and you'll notice the outside of the tire doesn't yet exist. Let's remedy this.

Go back to **NUM1** view.

Invert selection (**SPACE > Select > Inverse**)

Newer versions: **CTRL-I** for inverse selection

Press **FKEY** to create the missing faces, in the menu that pops up select **auto**

Newer versions: no auto necessary.

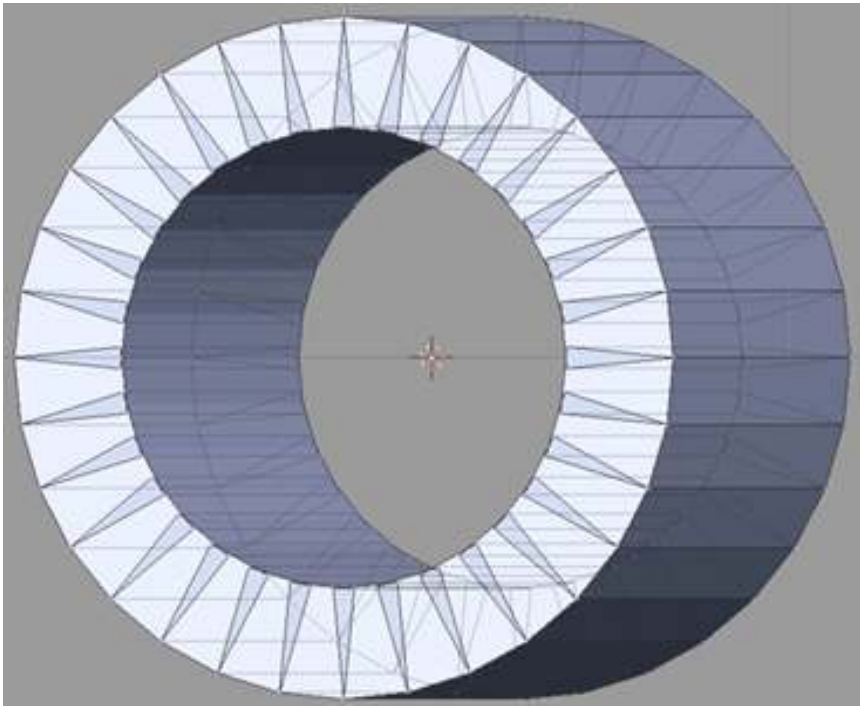


Figure 350 Checking the outer mesh

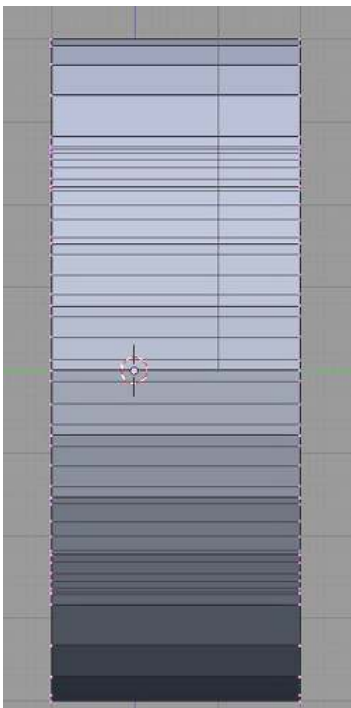


Figure 351 Reduce the width

55.2.4. Subsurf the tire

Now it's time to make the tire look like a rugged tire.

Apply a subsurf modifier of level 1 or 2 to the mesh. The tire will now look like a bead necklace. (*If not, remove doubles.*)

55.2.5. Crease the edges

A little creative use of creases will restore our tire.

Check that Limit Selection to Visible is off (*now called occlude background geometry on 2.49b*).

Enter **Edge Select** mode

Bring up the circle selection tool (**AKEY** to unselect all, then **BKEY** twice).

Use the scroll wheel to change the circle selection size to be in the center of the tire, between the inside and outside edges. This will select all of the inside edges, as well as the triangles on the side of the tire, as in the picture below.

Now press **SHIFT+EKEY** to Crease these edges, you can type 1.000, and press **ENTER**, or hold **CTRL** to pull in steps till you see 1.000 in the status bar at the bottom of the view window.

Noob note: DO NOT turn the 'limit-selection' tool on for the above selection, also you MUST be in *Edge Select* Mode or the triangles will not get selected.

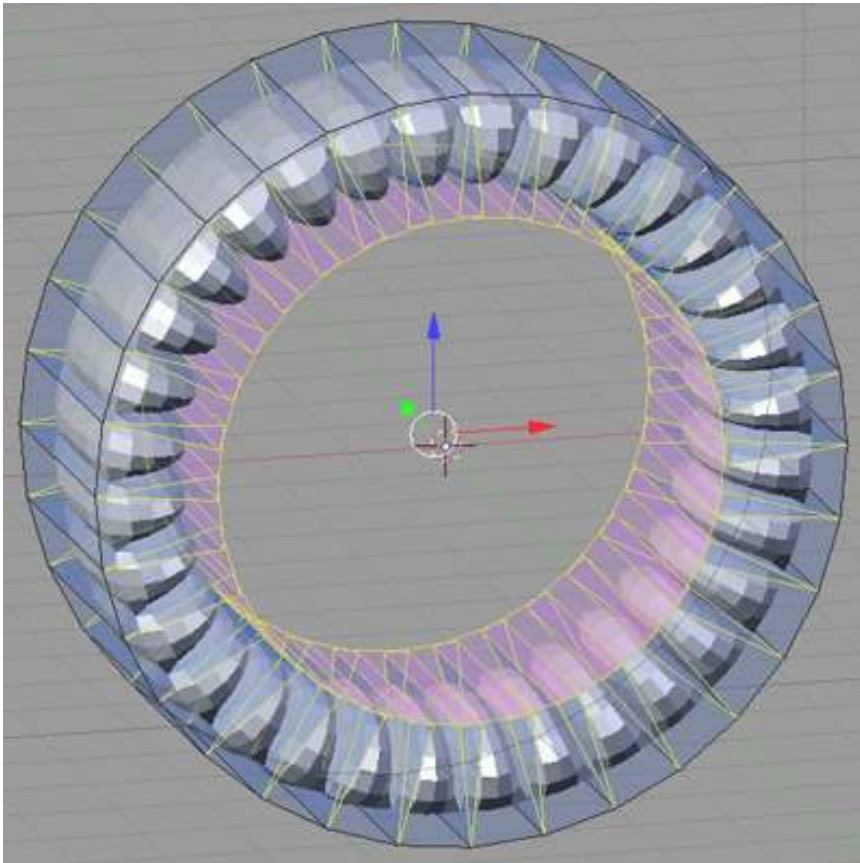


Figure 352 Applied subsurf

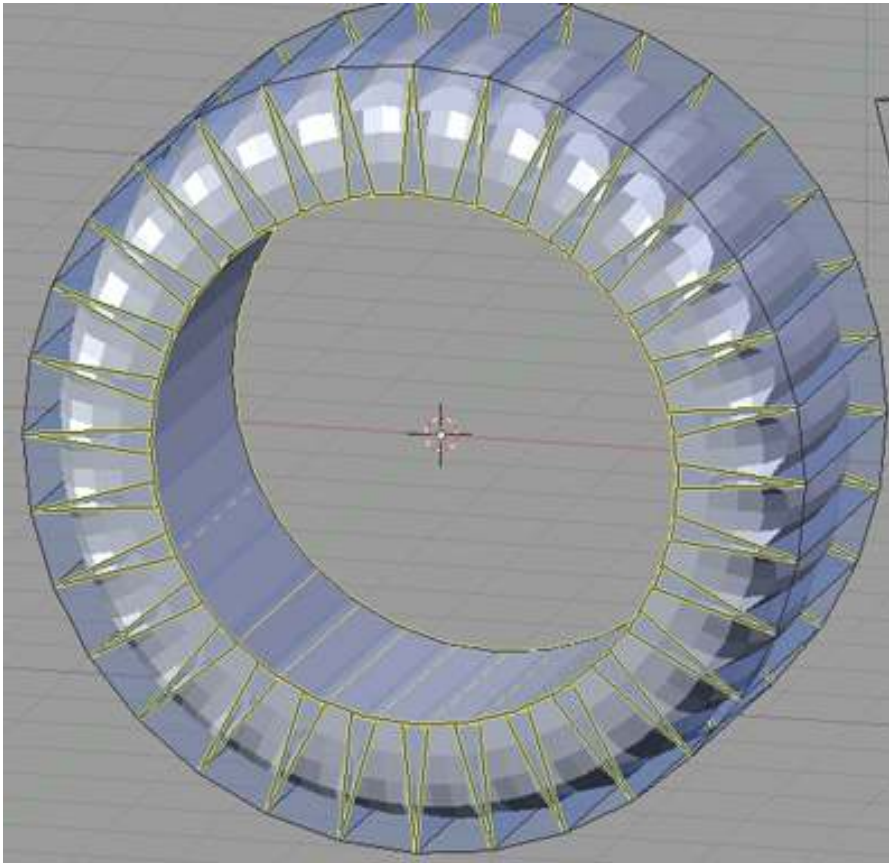


Figure 353 Crease the edges

55.3. Model the hubcap

The tire is almost done. Let's add a simple hubcap to it.

55.3.1. Create a cylinder

Be sure you're in **Edit Mode**

Hit **AKEY**, once or twice till the wheel is selected

Hit **NUM1** for front view

Press **SHIFT+SKEY**, from the popup menu, cursor->selection, to put the cursor at the center point of the existing tire

Hit space-> add-> cylinder with 32 vertices, radius of 1.9, depth of .5, and make sure the "Cap Ends" button is turned off,

Hit **NUM7** for top view

Hit **GKEY**, then **YKEY**, then type 2.2, and hit **ENTER**

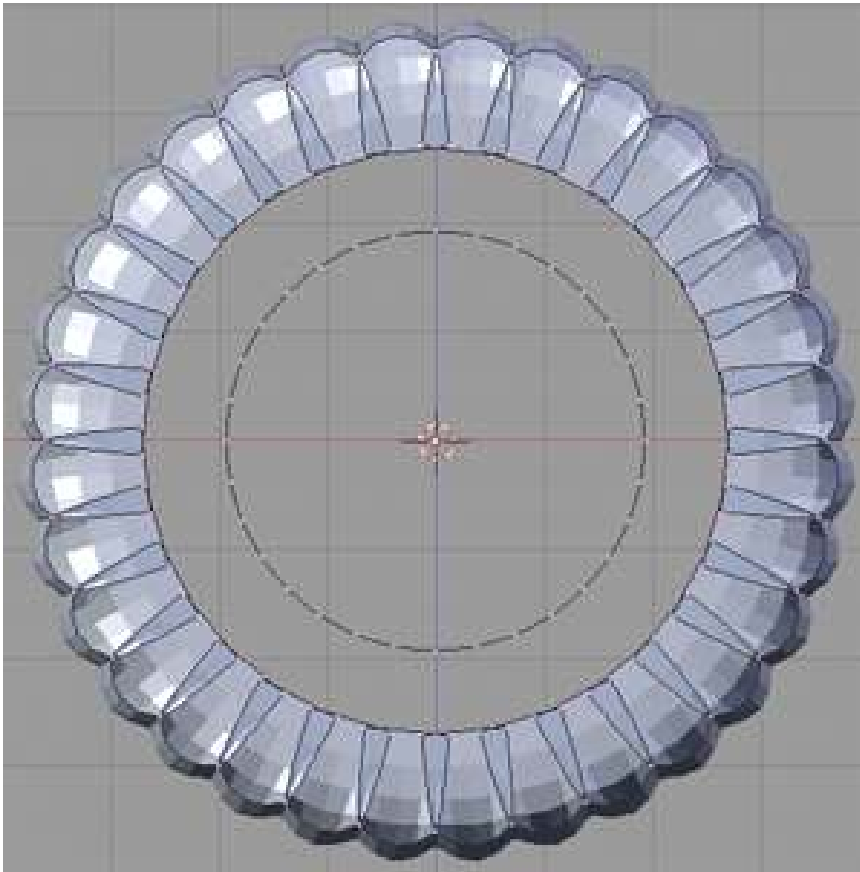


Figure 354 Add another tube mesh

55.3.2. Create the outside of the cap

Hit **AKEY** once, so that nothing is selected

Hit **NUM7** for top view

Zoom in with the **MMB** till the hub cap fills the view

be sure you're in **Vertex select** mode

Noob Tip: Make sure Occlude Background Geometry Mode is off

hit **BKEY** for box select, then holding the **LMB** drag the box along the top edge of the hubcap.

Hit **EKEY** on the popup select **Only Edges**, then hit **ESC**, to create the edges we will need

Hit **SKEY**, then **SHIFT+YKEY** to only move in the XZ axis, then type in **.35**, and hit **ENTER**

Hit **GKEY**, then **YKEY** to only move the Y axis, then type in **.35**, and hit **ENTER**

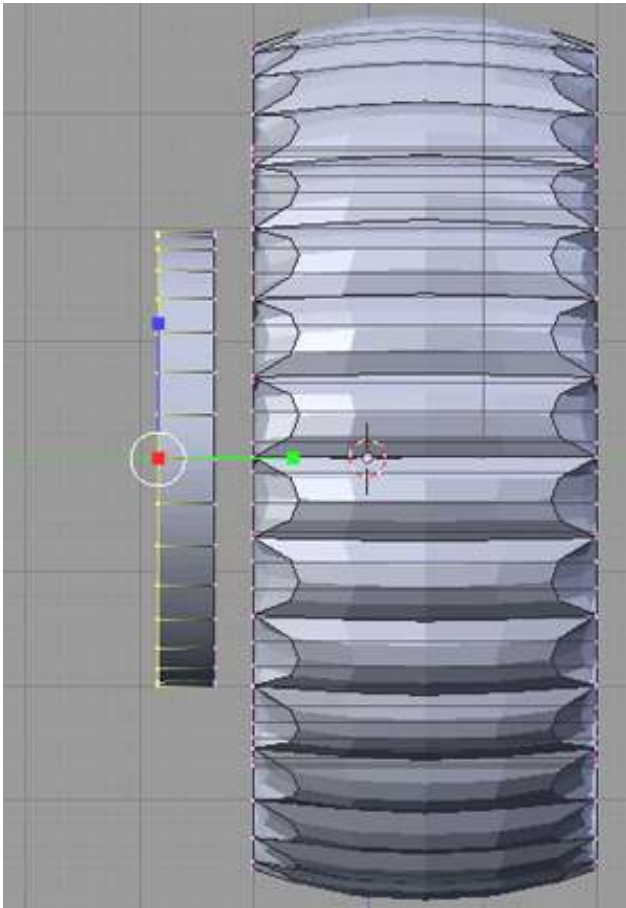


Figure 355 Shorten the hubcap width

55.3.3. Create the Axle Cover

We'll merge these vertices together to create a flat surface.

Hit **EKEY**, on the popup select **only edges**, then hit **ESC**, to create the edges we will need

Hit **ALT+MKEY** on the popup pick **at center**. Blender will reduce the 32 vertices to 1.

Hit **NUM3** for side view

hit **GKEY**, then **YKEY**, to only move the Y axis, then type in -0.4, and hit **ENTER**

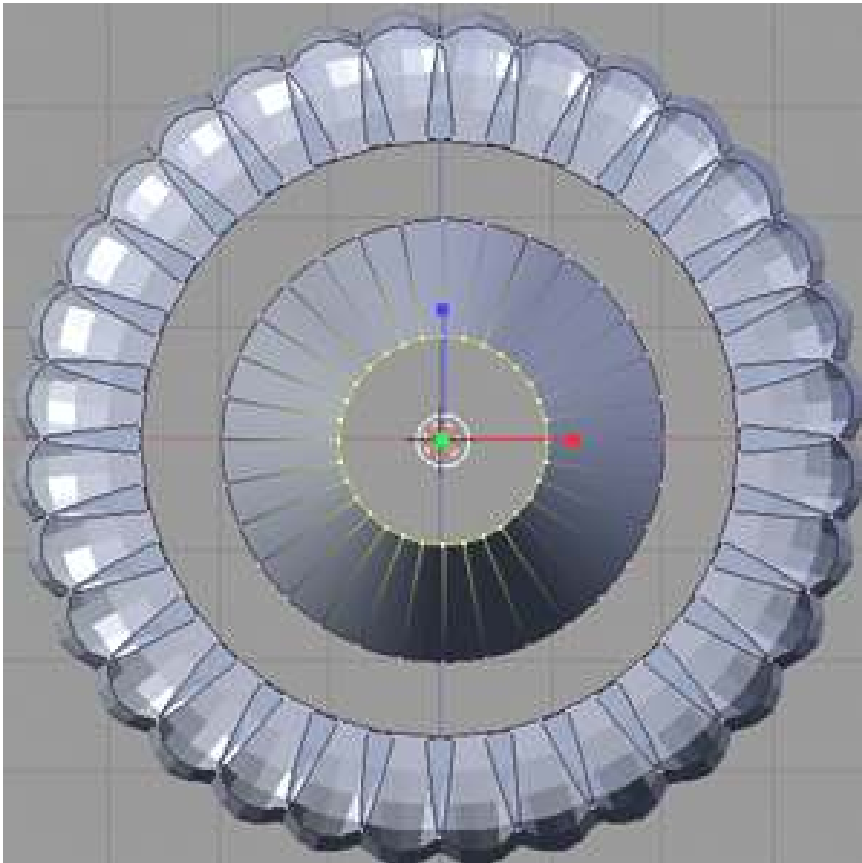


Figure 356 Scale in the front vertices

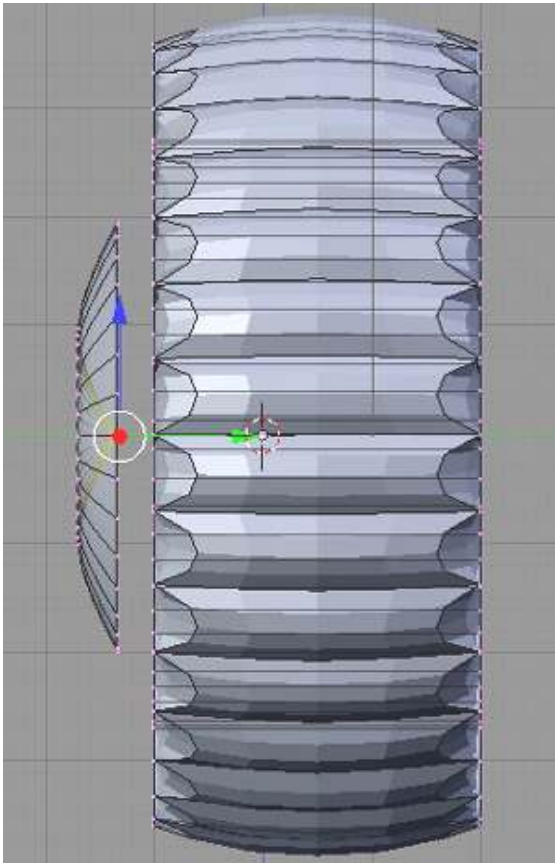


Figure 357 Pull the center vertex in

55.3.4. Final sizing of hub cap to tire

The final mesh editing is to scale the hub cap to a size that is slightly larger than the hole of the tire.

hit **NUM7** for top view

position the mouse over the hubcap, and press the **LKEY** to select the entire hubcap

Hit **SKEY**, then **SHIFT+YKEY** to only move the XZ axis, then type 1.48, and hit **ENTER**

Hit **NUM3** for side view

Hit **GKEY**, then **YKEY**, to only move the Y axis, then type -1.11 (use -0.77 if you want your hubs sticking out), and hit **ENTER**

55.3.5. Renaming the Wheel

The last thing to do is to rename the wheel so we can find it easier later.

Enter Object mode and select the wheel only.

In the Link and Materials subwindow in Editing (**F9**) you'll see the Active Object name box. It should read **OB:Tube** in the grey box. This name was created because we started with a tube mesh. In newer versions, it will be **cylinder** instead of **tube**.

Click on the button and rename the object to something like 'wheel'. Save your file where you'll find it later and continue to the next step.

Noob note: although not something you should normally do, if you wish to create separately named objects for the hubcap and the wheel you could initially create them both separately in object mode, or split objects in edit mode with **P** (seParate) or in object mode, join objects with **CTRL+JKEY** (can cause side effects).

55.4. Extra



Figure 358 Rendered tire

Change the materials to make it look like a tire. One object can have multiple colors/textures, as you will learn in future tutorials.

If you'd like to race ahead, then refer to the materials¹ section for an explanation on how to do that, or to the Blender manual: Multiple Materials²

¹ Chapter 36 on page 213

² http://wiki.blender.org/index.php/Doc:Manual/Materials/Multiple_Materials

56. Simple Vehicle: Seat

56.1. Techniques

You should already know how to:

- Make a mesh
- Navigate the viewport
- Extrusion
- Subsurf
- Crease edges

This section will recap and introduce:

- Loop subdivide
- Small, consistent vertex movement

The design will be an all-terrain bucket-type seat.

56.2. Extrude the Seat

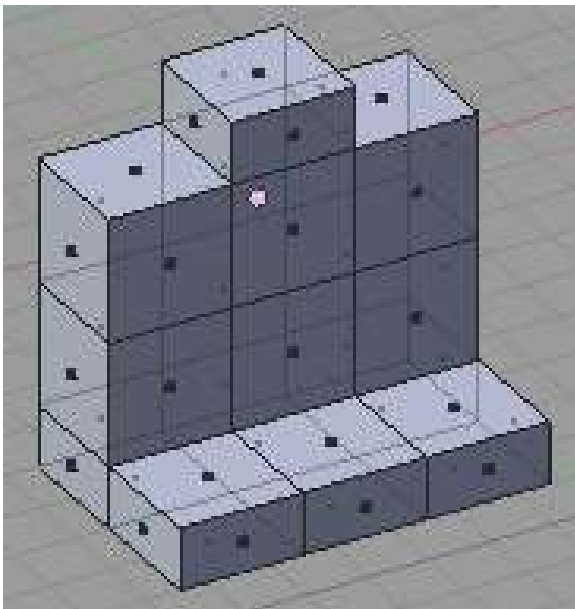


Figure 359 Basic setup of shapes.

Start in **NUM 1** view of the default cube and rename it.

Extrude the cube multiple times to make your basic shape. In this example a 3x3x1 block composes the body with one cube coming out the top for the headrest, and the bottom cube's front faces extruded out to create the seat.

Noob Note: best practice is use "Face" selection mode, select the face, or faces you want to extrude with the **RMB**, then hit **EKEY**, select an axis to move on by pressing the **XKEY**, **YKEY**, or **ZKEY**, and pull with **CTRL** held down.

*(If you insist on using **Edge** select mode, or **Vertex** mode, you want to only pick "Faces", from the popup menu that appears, after you hit the **EKEY**; as this will stop the doubles, and inside faces, from appearing. If you don't see **Faces** on the popup menu then you are really extruding from doubles with reversed normals, so go and remove them (**WKEY** -> remove doubles), and possibly recalculate the inner and outer normals, before continuing)*

56.3. Add cushion seams

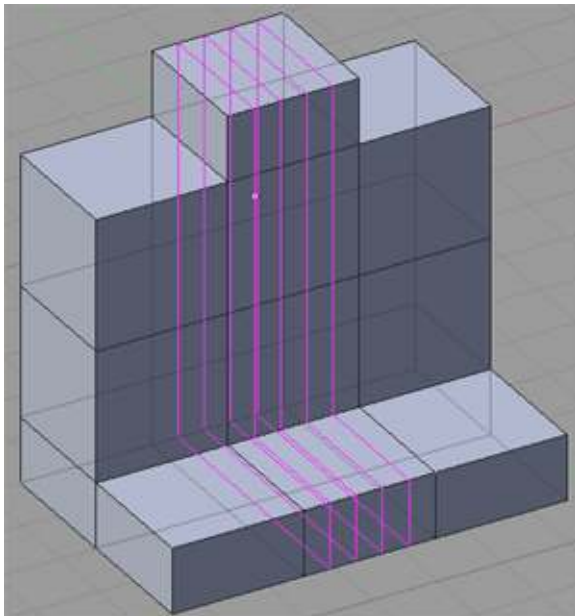


Figure 360 Multiple loop subdivide.

Noob Note: Before starting on this, you'll need to make sure you don't have any unnecessary faces **inside** the seat, or else you'll get strange cushion seams.

To add a little texture to the mesh, we'll add some cushion seams. Use Loop Subdivide (**CTRL R** or **K** > Loop Cut) and you'll see the pink selection loop. You can use the **NUM+** key to increase the number of loops made at the same time. Use mouse wheel or press **NUM+** 3 times to form 4 loops and **LMB** the center column of blocks. You may also find it easier to add them one at a time in the correct place, than inserting them and then moving them.

(To get multiple Loops instead of pressing + just press the number of loops you want in this case "4", this is a fast easy way to achieve this.) You can use a mouse-wheel as well.

Noob note: If the loop comes up with green lines rather than purple you have gone one step too far, just press **esc** (escape) and try again. When you see the purple lines use your **Mouse Wheel** or the **NUM+** button.

Version 2.60 user comment: I tried several different edge loop, loop cut, loop subdivide, etc. to no avail, so I selected all the edges of the middle portion of the object (seat) using box select, then I Knife Cut (hold down Kkey and LMB through the front view of the edges from top to bottom. Select multi-cut and number of cuts to 4.

56.4. Position the cushion seams

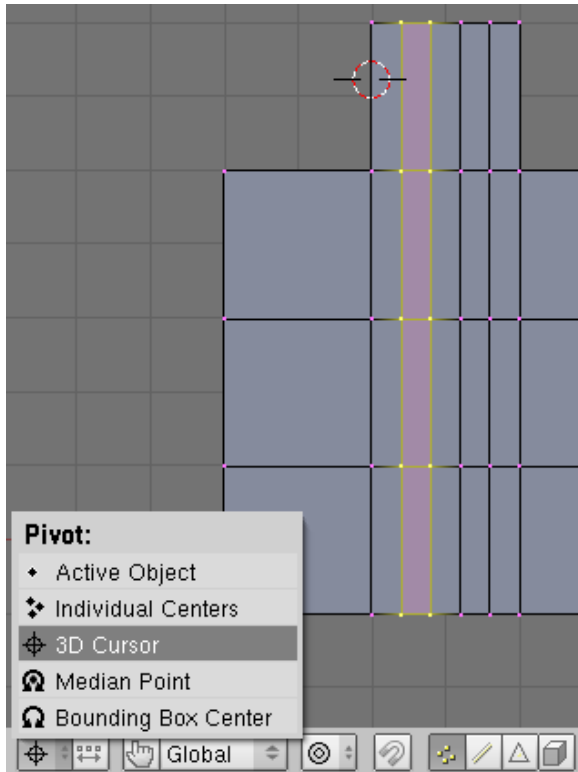


Figure 361 Where the 3D cursor should be placed and what to select

Go into ZX view (**Num 1**) and make sure the view is orthographic (**Num 5** to activate/deactivate orthographic view). Place the 3D cursor on one of the sides of the head rest (**LMB**, or **SHIFT+S** to move cursor).

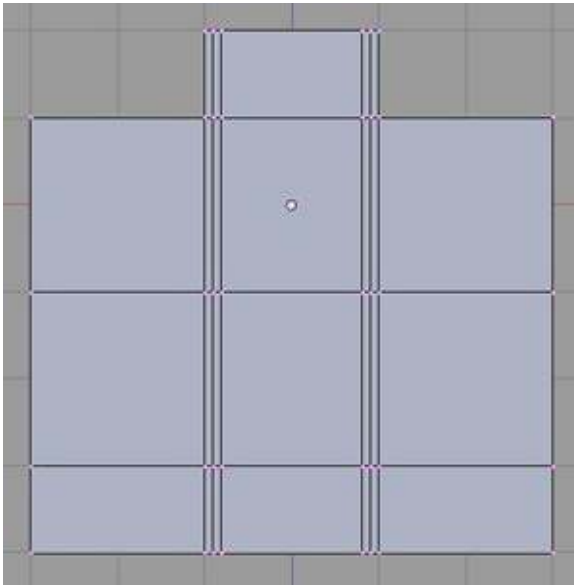


Figure 362 Positioning the seams.

Set the pivot to 3D cursor and select the two closest loop cuts. Scale (**SKEY**) it down on the X axis (**XKEY**) to 0.3. The goal will be to have the loop closer to the cursor to go into the cushion to become a seam. Now, do the same thing for the other side of the head rest.

Noob note: The easiest way to select the two closest loop cuts is to first select (**RMB, SHIFT+RMB**) one edge of each loop, and then go to Select Edge Loop (**SPACE -> Select -> Edge Loop**). Or hold down **ALT** when selecting one of the loop's edges. This should select the entire loop. In order to select more than one loop, hold down **SHIFT** as well.

56.5. Add Depth to the seams

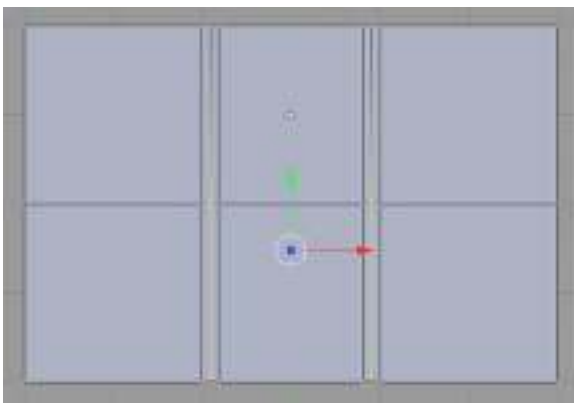


Figure 363 Vertical seam creation.

In overhead view (**NUM 7**), select the vertices in vertical parts of the two front facing seams of the seat back, grab them (**GKEY**), move only along the Y axis (**YKEY**), and type in **.05 ENTER**.

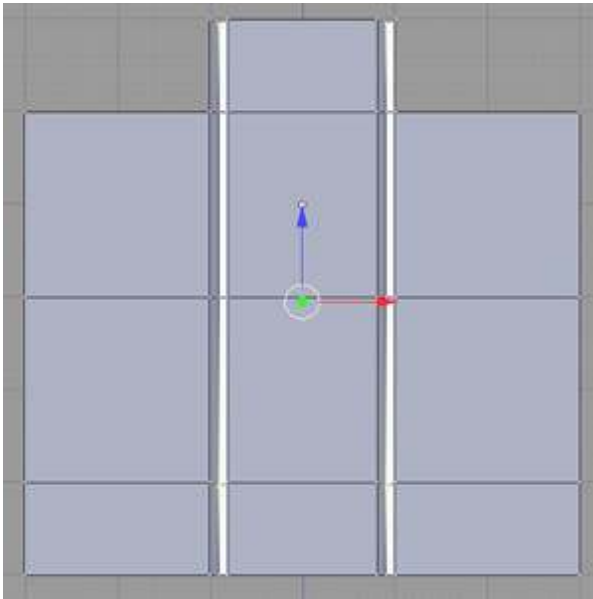


Figure 364 Horizontal seam creation.

Switch to **NUM 3** view and move the vertices in the horizontal parts of the same two seams, grab them and move them down by moving them -0.05 along the Z-axis.

Noob Note: I've found that an easier way to do this is selecting each loop of the seam one by one and scaling to $.95$ (make sure the pivot is set to "Bounding Box Center")

56.6. Subsurf the seat

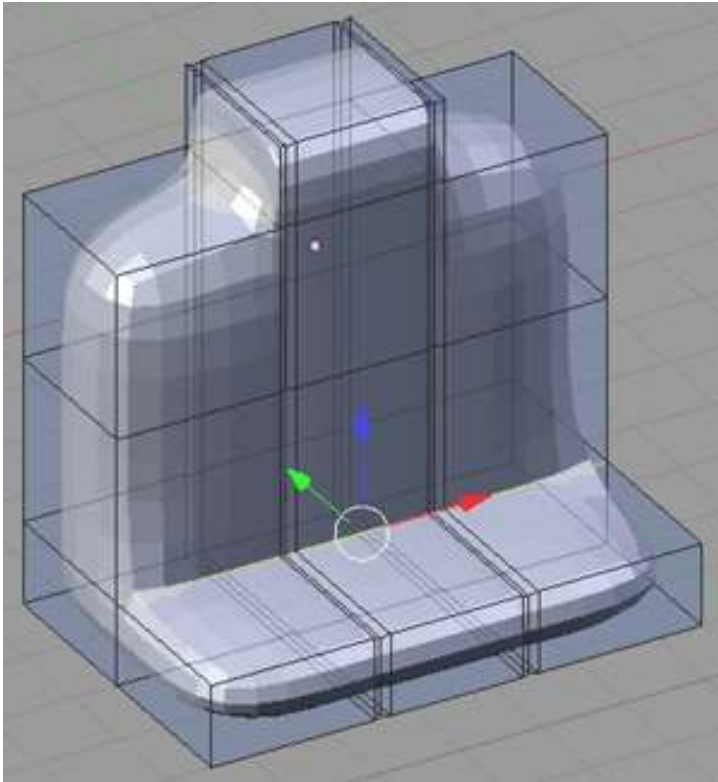


Figure 365 Subsurf seat

Add a subsurf modifier to the object. Select the edges between the back and seat and crease (**SHIFT+EKEY**) them. Crease any edges you feel like to create your perfect jeep seat.

Noob note: it's best to be in **Edge** select mode when creasing

A subsurf level of 2 or 3 looks best, and don't forget to change the render level to 3 or 4.

Select all(**AKEY**) then either hit the "Set Smooth" button at the bottom of links and materials, or hit **WKEY** and select "Set Smooth" for a much smoother subsurf.

Noob note: If your seat is noticeably misshapen after adding the subsurf modifier, you may just have to delete internal faces in your model. **TAB** into edit mode, and hit **ZKEY** to get into wireframe mode. Click the face select button and look for faces that are totally inside the model. There will probably be a couple vertical faces (in YZ-plane) under the seams in the seat. I found a few elsewhere, also. Deleting all these cleared everything up.

Noob note 2: An other way to remove the extra faces (which can cause the "seams" to be very deep) is to go into edit mode, select your whole seat (**AKEY**), then hit (**Wkey**) and select "Remove doubles" from the menu. This is much quicker than finding them individually, and should solve the problem.

Noob note 3: The problem may also be solved by going into edit mode, using **AKEY** to select all, and using **SPACE** > Edit > Normals > Recalculate Outside (**CTRL + NKEY**).

56.7. Resize the seat

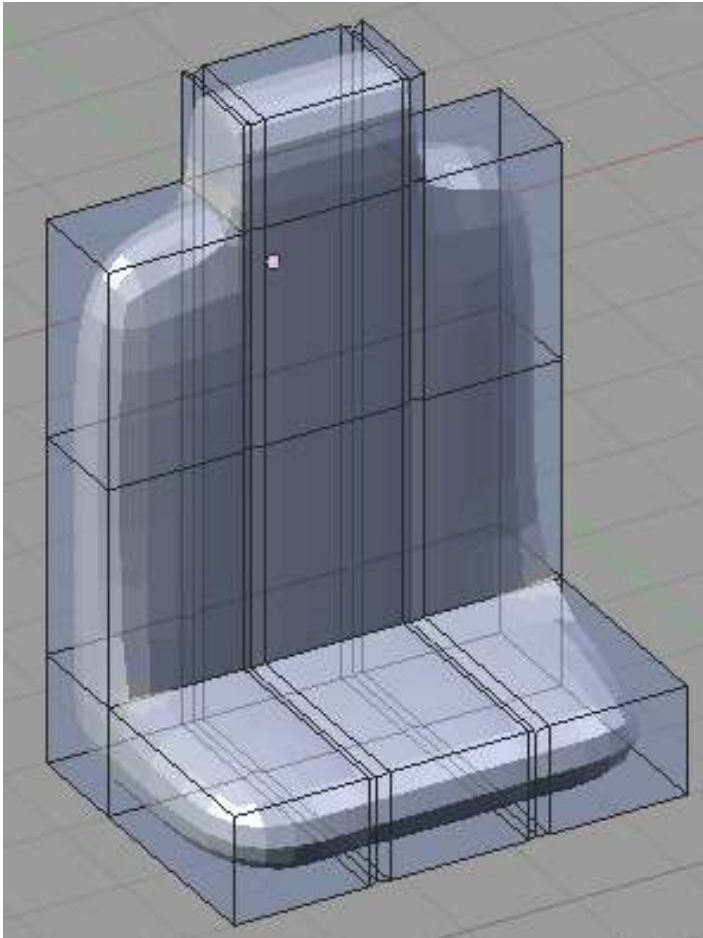


Figure 366 Change the widths.

Next, resize the seat's height and width.

Note: Be sure to change the rotation/scaling pivot back to center point!

To make the whole seat narrower in width, select all **AKEY** twice, then hit **SKEY**, followed by **XKEY** then type 0.8 and press **ENTER**.

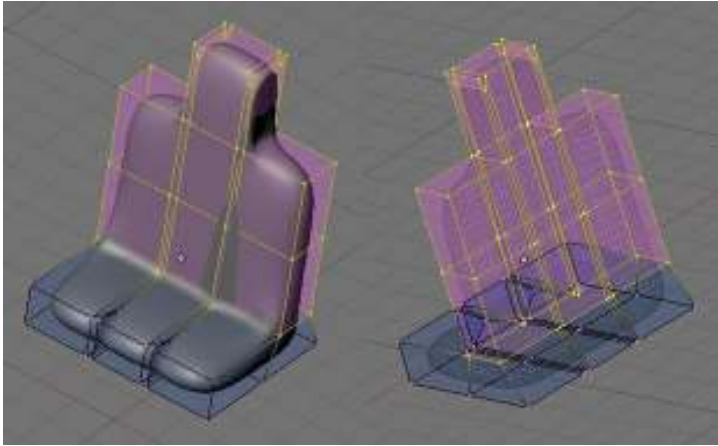


Figure 367

To make the seat back a bit narrower in thickness, select around the seat back (in vertex mode) with the circle select **BKEY, BKEY**. Once you have it completely selected on all of the sides, hit **SKEY**, lock axis with **YKEY**, and type 0.75, and hit **ENTER**

56.8. Final touches

This final seat renders to:



Figure 368 Rendered seat.

56.8.1. Leathery look

To give the seams a leathery cord look, Hit **AKEY** twice to select all, then **WKEY** and choose Subdivide Fractal on the popup menu; just keep the defaults and the seams will look like a bunch of vines until you render it and they look like leather seams .

56.8.2. More concave

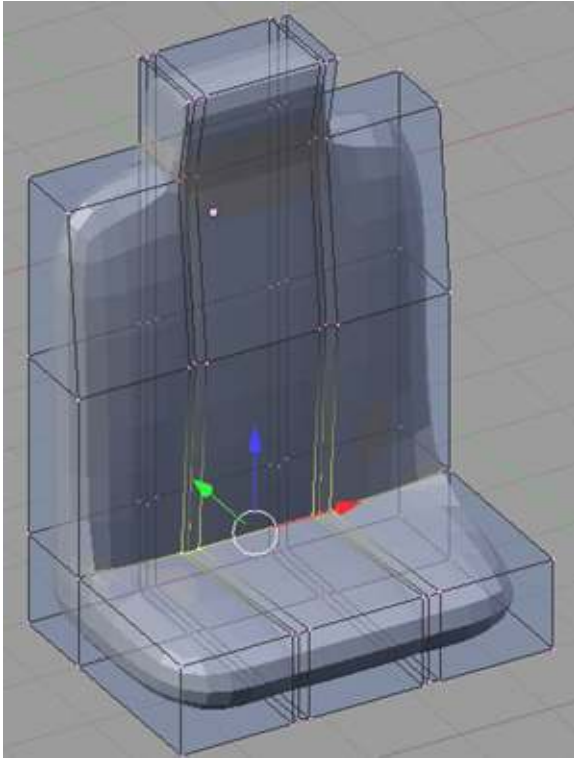


Figure 369 Making it concave.

Also the seat can also be made slightly more concave to look like it would hold a person better.

57. Simple Vehicle: Rocket Launcher

Blender3d Aligned to view issue

This tutorial relies on objects being created so that they are aligned to the view that you're looking through. Versions 2.48 and above have changed the way this works. Visit Aligned (http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Aligned_to_view_issue) to view issue to understand the settings that need to be changed.

57.1. Techniques

You should already know how to do:

- Previous Simple Vehicle techniques

This section will recap and introduce:

- UVspheres
- Changing object's center

57.2. Overview

Two assumptions are going to be made here. One is that the rocket will not be launched in the future (use separate objects if you want to do that). The other is this is going to be a simple, simple design.

If you want to add options to your gun (think sight, trigger), go for it!

57.3. Create the Launcher

Start a new file and delete the default cube.

57.3.1. Add a cylinder

In **NUM 1** view, add a cylinder mesh with 24 vertices with End Cap button **pushed in** (or *ON*). We'll use 24 because the default 36 is overkill and will only increase rendering time. Rename and elongate the cylinder along the Y axis. This will be the length of the launcher (minus the rocket).

Noob note: if your cylinder shows up facing the wrong direction, then see the Blender_3D:_-Noob_to_Pro/Aligned_to_view_issue¹ page, that describes how to fix this problem with 2.48 and up.

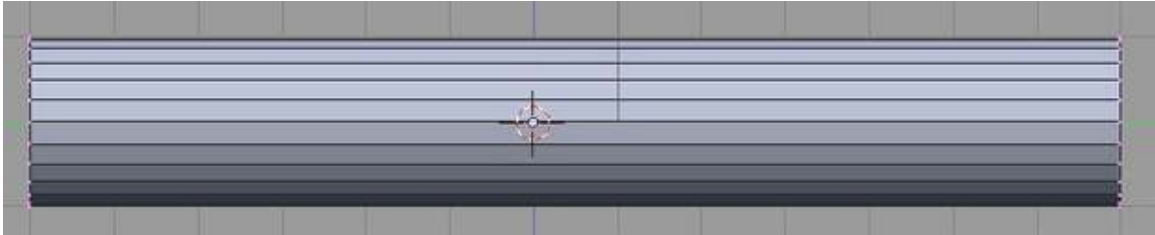


Figure 370 Elongated cylinder

57.3.2. Extrude a tube from the right end

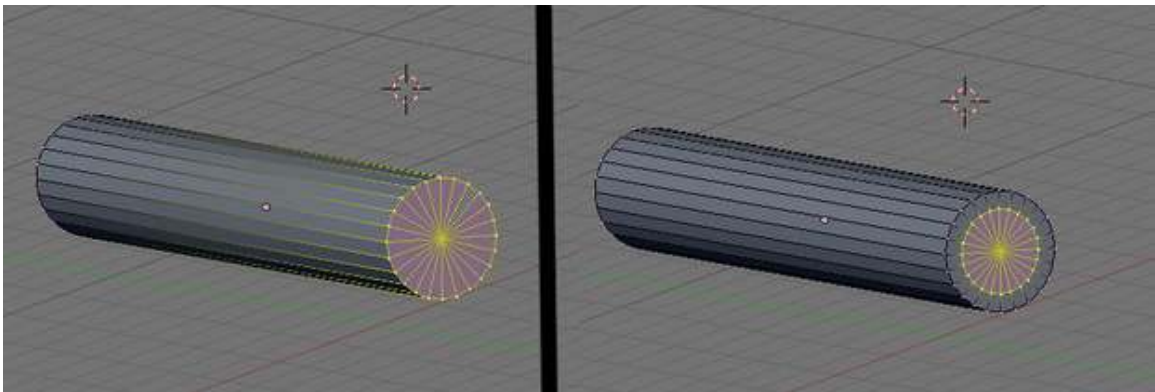


Figure 371

Circle Select (**BKEY**, **BKEY**) the vertices at the right end, then extrude (**EKEY**) them as region, and press **ESC** to create a copy of the vertices.

Scale (**SKEY**) them by 0.7.

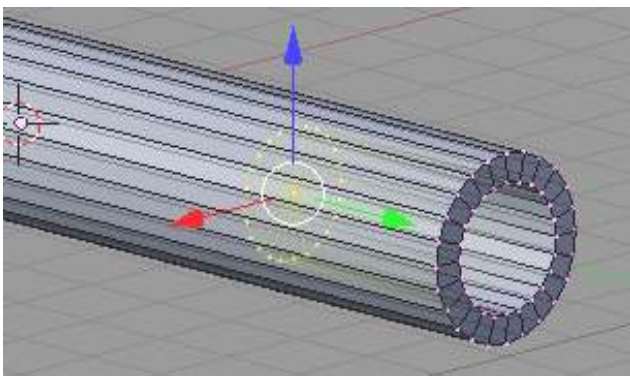


Figure 372 Invert the cylinder end to make a tube

¹ Chapter 42 on page 251

Extrude (**EKEY**) them as region toward the left, along the Y axis (**YKEY**); This will make the rocket launcher look like it's a tube.

Note: For **older** versions, which **do not create end caps** keep the previously extruded edges still highlighted. Then extrude (**EKEY**) them again, and then hit **ESC**, now merge the vertices at the center (**ALT + MKEY**). If you did this correctly, 23 vertices should be removed, and all come together to form a group of triangles inside of the tube.

57.3.3. Extrude a tube from the left end

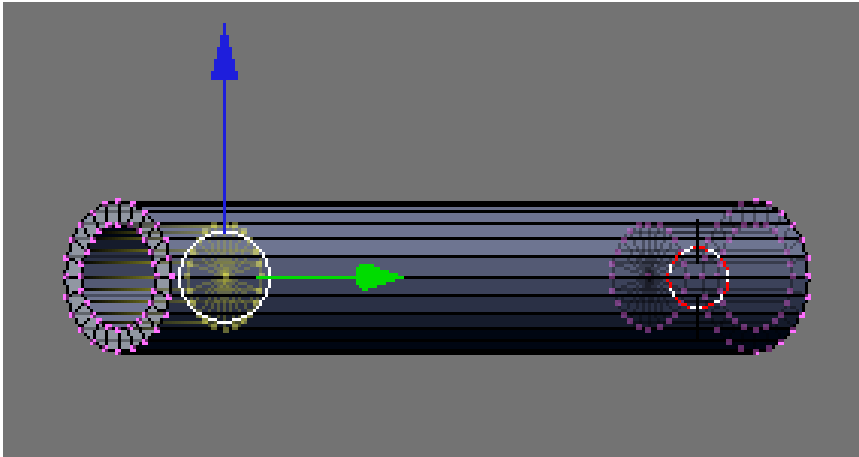


Figure 373 left end

Now work on the left end, by following the same steps used for the right end.

Noob Note: before starting on the left end, be sure to hit **AKEY** twice to deselect the internal face you just created on the right side, which you can't see, unless you hit **ZKEY** to show wireframe mode.

57.4. Create the rocket

For the purposes of this tutorial we will add the rocket on the left end of the launcher.

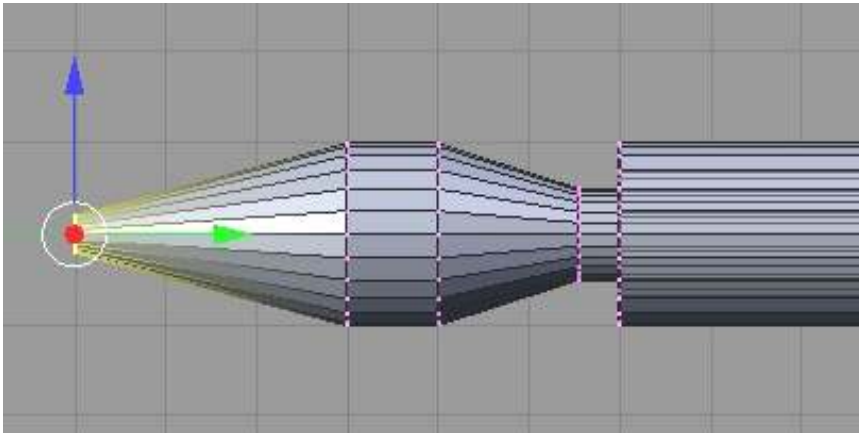


Figure 374 Finishing the rocket head

57.4.1. For a one piece rocket + launcher

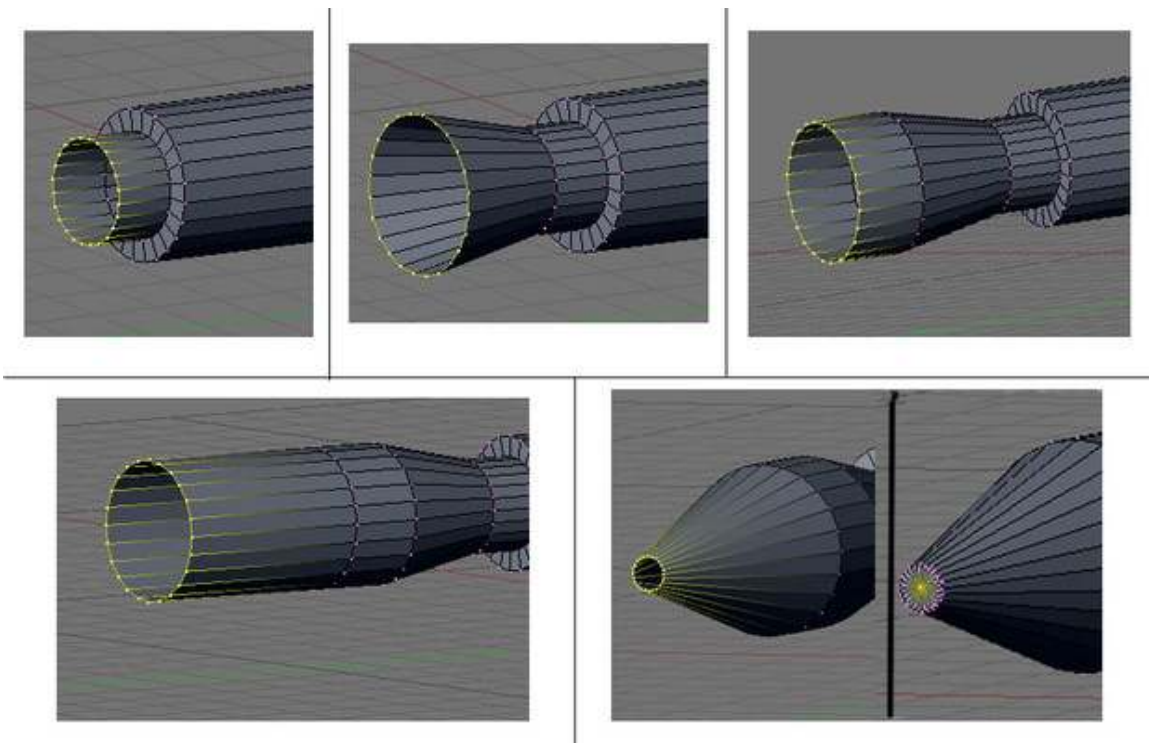


Figure 375

Select the inner ring of vertices using circle select (**BKEY** twice)

Extrude (**EKEY**) the "edge" along the Y axis (**YKEY**) twice, then scale the new edge by 0.8.

Extrude the "edge" along the Y axis, and scale the new edge by 1.5.

Extrude the "edge" along the Y axis, and scale the new edge by 0.2.

extrude the "edge", hit **ESC**, and merge (**ALT+M**) at the center to form the face for the nose

57.4.2. For a two piece rocket and launcher

- **Method 1:** create a cone, rotate it, then extrude and scale to get the rocket shape.
- **Method 2:** create a cylinder, scale it along the Y-axis, then extrude one side, scale to about 1.3, then extrude two more faces, scaling the first about 0.3 and merging the vertices of the second at center.
- **Method 3:** Create a UV sphere with 4 rings, and model it - much in the same way as the penguin - into a rocket.

57.5. Create the mount

Having a launcher is nice, but we'll need to affix it to the jeep somehow. Let's add a mount to the tube.

Make sure you're in edit mode, not object mode!

57.5.1. Add a cylinder

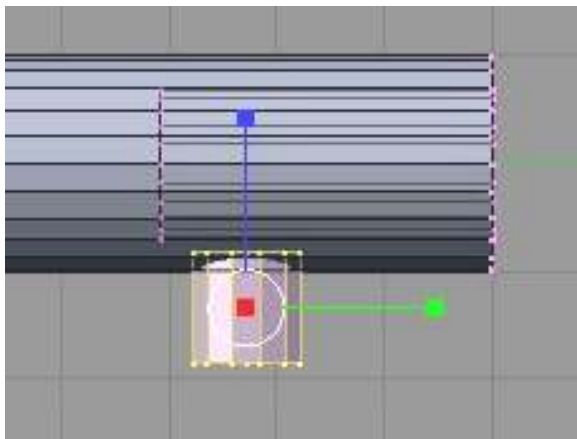


Figure 376 Mount arm

Add a cylinder with 24 vertices with End Cap button pulled up (or OFF), since we won't be seeing the ends.

Scale with the Z axis locked, to about 0.55, then move it to the bottom of the tube.

57.5.2. Add a UVsphere

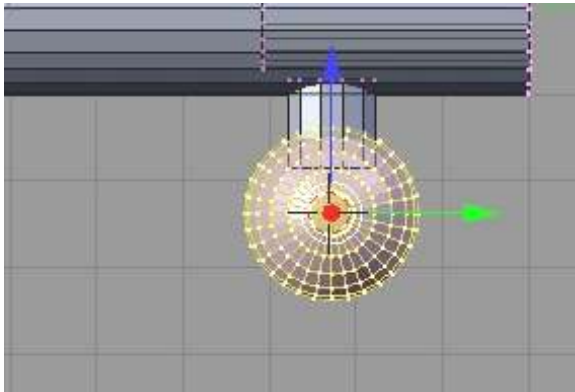


Figure 377 Place the sphere

The easiest way to have a wide range of motion for the launcher is to use a ball joint. We can simulate one by just adding a UVsphere. The default 32 segments and 12 rings will be fine. This creates a smooth sphere.

You can think of the number of segments as being the wedges visible when the sphere is viewed as you added it. The number of rings then could be described as the depth of the sphere from that same view.

Resize and place the sphere at the new cylinder arm.

57.5.3. Reposition the center point

This next step will be important for continuing the tutorial. Get the 3D cursor to as close to the center of the sphere as possible. While the sphere is still selected after creation, you can press **SHIFT + SKEY** and snap cursor to selection, putting it in the exact center of the UV Sphere.

Switch to Object Mode. Under either the Object menu or **SPACE** go to Transform > Center Cursor. This should move the large pink dot where the cursor is located. This will give it a new center of gravity around the ball joint, making it easy to manipulate later.

Noob Note: If you have already deselected the sphere, press **LKEY** while hovering your cursor over the sphere. Make sure not to include any vertices from the attached cylinder).

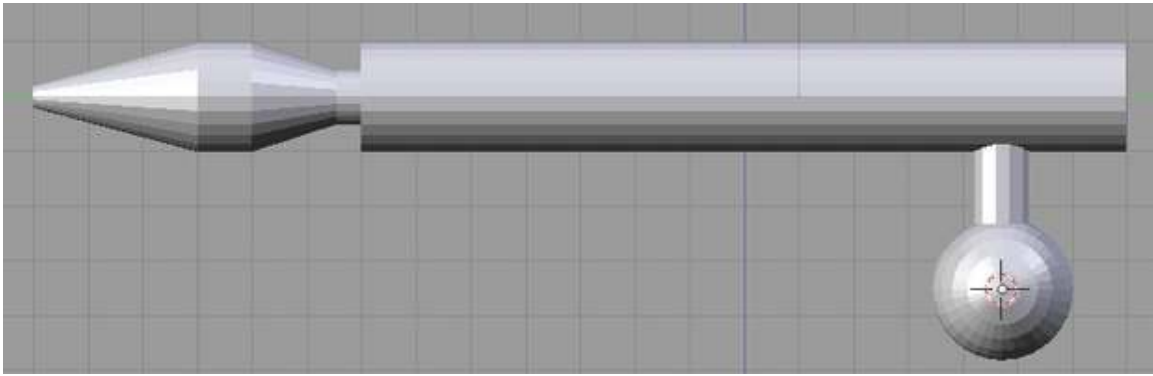


Figure 378 Change the center of the object

57.6. Subsurf

Be in **edit** mode!

Apply subsurf, level 2

Select everything (**AKEY** twice), and hit the **set smooth** button (at the bottom of the links and materials panel).

This makes the ends of the rocket launcher tube too rounded, as real ones are squared up.

57.6.1. Square up the tube edges on the right side

On the right side (without the rocket sticking out) be in **Face** select mode

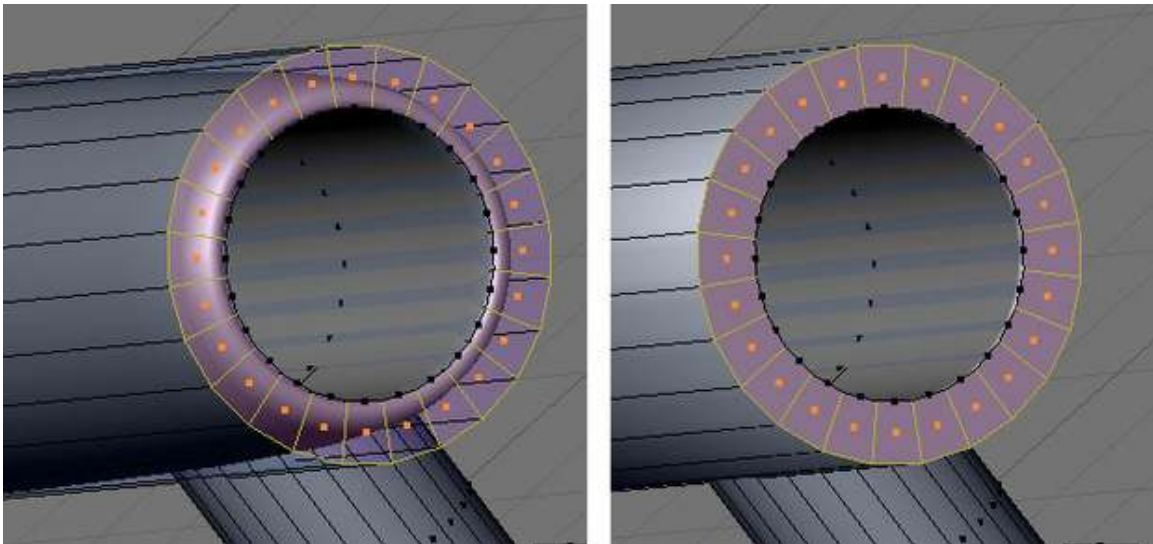


Figure 379

RMB on one of the faces in the outer ring, then circle select (**BKEY** twice), and select all the faces at the end of the launcher tube. You can also do a face loop selection by selecting a perpendicular edge to the loop with **ALT+RMB** in face selection mode.

Hit the **set solid** button (at the bottom of the links and materials panel).

Crease the edges of the faces by hitting **SHIFT+E** and set it to 1.0.

57.6.2. Squaring up the tube edges on the left side

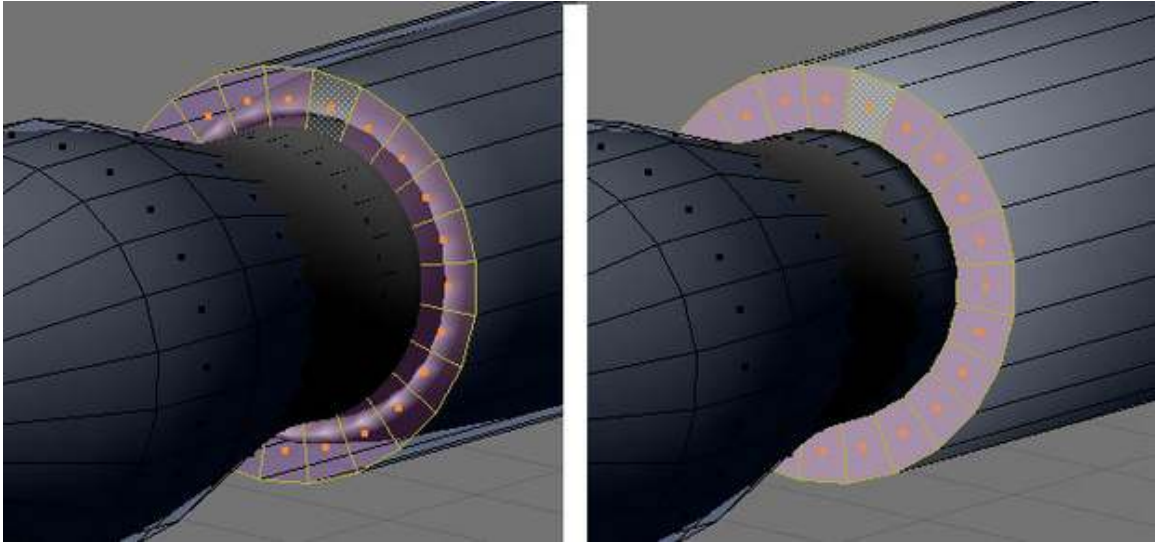


Figure 380

Now do the left side exactly like the right side.

Noob Note Be sure to deselect everything on the other side first (**AKEY** twice)!!

57.6.3. Squaring up the rocket

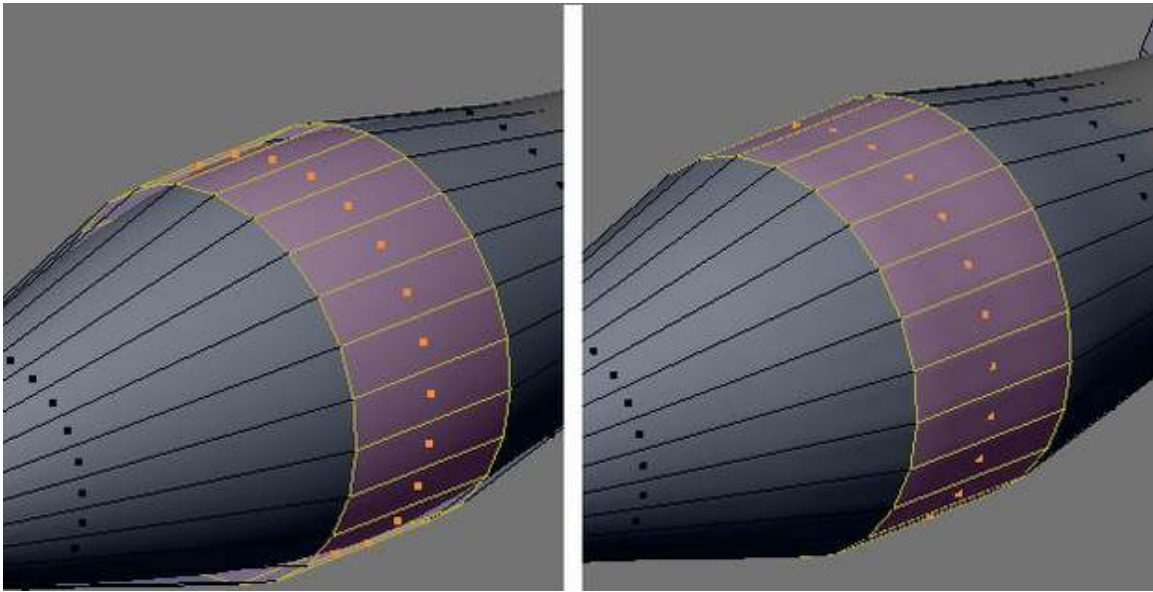


Figure 381

RMB on one of the faces in the middle surface, then circle select (**BKEY** twice), and select the faces all the way around

Crease the edges of the faces by hitting **SHIFT+E** and set it to 1.0.

57.7. Final touches

Apply materials or additional items to the object and save for later use.

Do not forget to name the rocket for later use.

58. Simple Vehicle: Body

58.1. Techniques

You should already know how to:

- Make a mesh
- Navigate the viewport
- Extrusion
- Form faces
- Name objects

This section will recap and introduce:

- Deleting and creating edges
- Subdividing
- Merging vertices
- Loop subdivide
- Adding unconnected vertices in one object

58.2. Planning

The jeep is being designed to include the back, flatbed, door holes, dashboard, window, and hood. The window is extruded straight up (older jeeps' windshields aren't slanted), and I decided to add a lower back to later hold the bumper/lights if you wanted to add them.

58.3. Building the Jeep

58.3.1. Extrude the Chasis

Start a new file.

Change to front view (**NUM1**), switch to **EDIT** mode, and deselect all (**AKEY**).

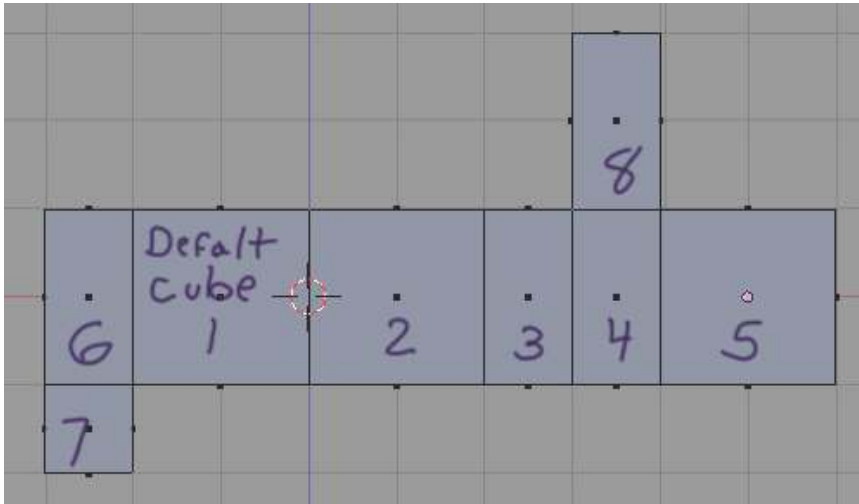


Figure 382 Main sections are made.

Be in **Face** select mode

Starting from the default cube (#1), box select (**BKEY**) on the right edge to select the right face, and extrude (**EKEY**) 2 units (#2) to the right along the X axis (should default, otherwise **XKEY**). Continue with three more extrusions to the right along the X axis of 1 unit (#3), 1 unit (#4), and 2 units (#5).

deselect all (**AKEY**).

Now starting again from the default cube (#1), box select (**BKEY**) on the left edge to select the left side face, and extrude 1 unit (#6) to the left in the X axis

deselect all (**AKEY**).

box select (**BKEY**) on the bottom edge of block #6 to select the bottom face, and extrude 1 unit (#7) down in the Z axis.

deselect all (**AKEY**).

Finally box select (**BKEY**) on the top edge of #4 the second block from the right to select the top face, and extrude up 2 units (#8) in the Z direction.

58.3.2. Widen the chassis

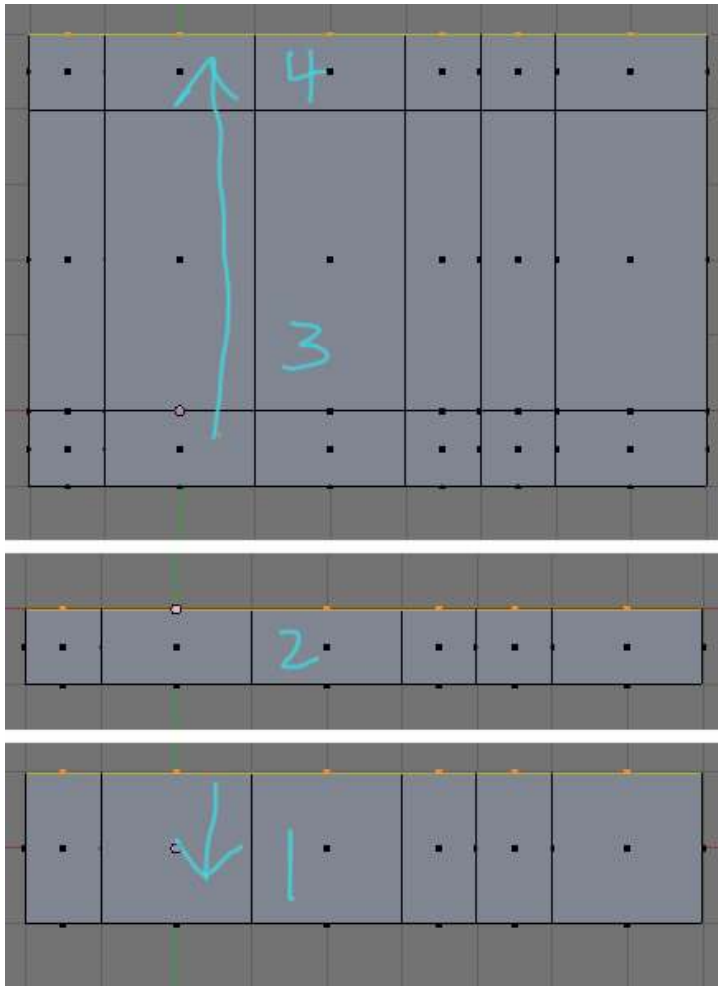


Figure 383

Now to widen the jeep body.

Switch to top view (**NUM7**).

deselect all (**AKEY**).

Reduce the width of the existing boxes (**#1**) to one square (**#2**) by box selecting (**BKEY**) the top edge (since you're in **face** select mode it will actually select all the faces on that side), now move (**GKEY**) 1 unit down in the Y axis (**YKEY**)

Hit **WKEY**, and select "remove doubles" from the popup (*2 will likely be removed*)

Extrude (**EKEY**) by "Region", the top side, up along the Y axis (**YKEY**); once for 4 units (**#3**), and then extrude again by "Region" for 1 unit (**#4**).

This should give you the figure that is shown to the right.

Noob note: holding **CTRL** as you do the above moves and extrusions will lock it to grid steps

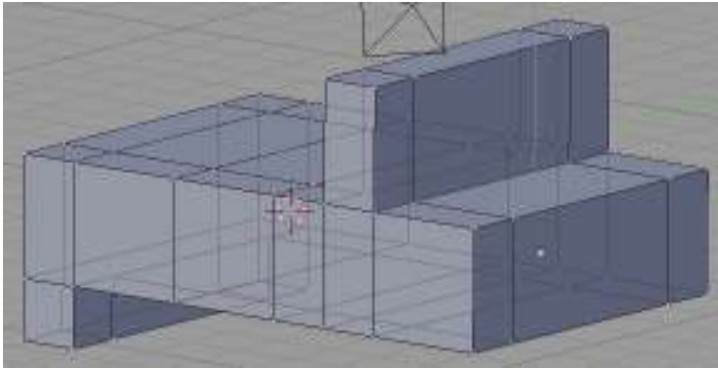


Figure 384 Widening along the Y axis

58.3.3. Flatbed and Doors

Our jeep design will have somewhat of a cheat - no actual door.

There are now a few different ways of doing this (thanks to reader submissions)! Try each one and study the results, as they will teach you about the issues you will encounter when you start making models of your own design;

Method 1 is the simplest of them, but teaches you the least in dealing with mesh trouble.

Method 1

Make the jeep bed

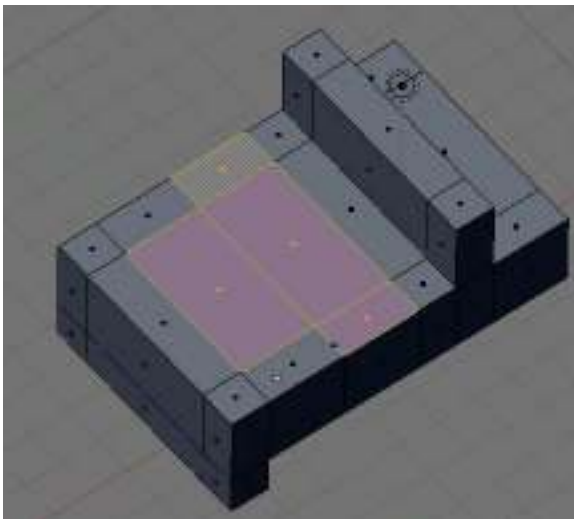


Figure 385

In NUM7 view

Be in **Face select** mode

Turn **occlude background geometry** on (the box to the right of vertex, edge, and face select mode).

Select (**RMB**, then **SHIFT+RMB**) the top faces where the flatbed and then doors should go. You should now have four faces selected, two large ones for the bed, and two small ones for the doors.

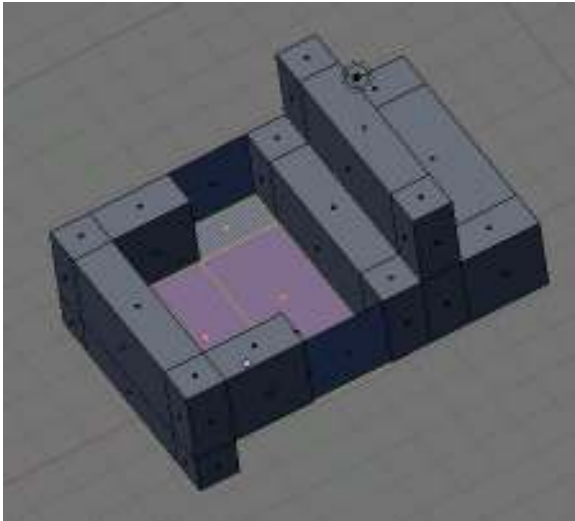


Figure 386

Extrude (**EKEY**) the "region" -2, on the Z axis (**ZKEY**).

Make the door holes

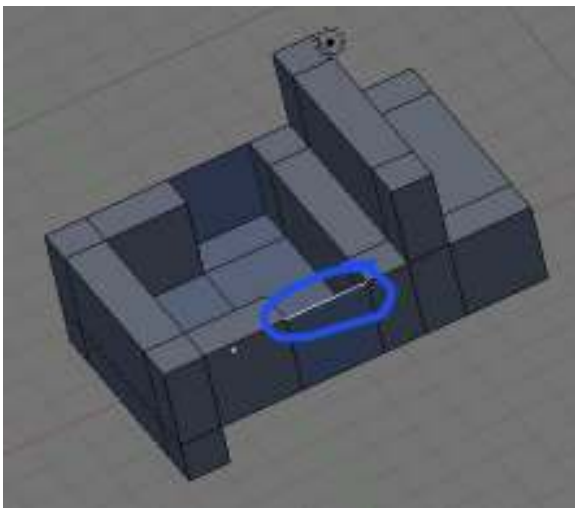


Figure 387

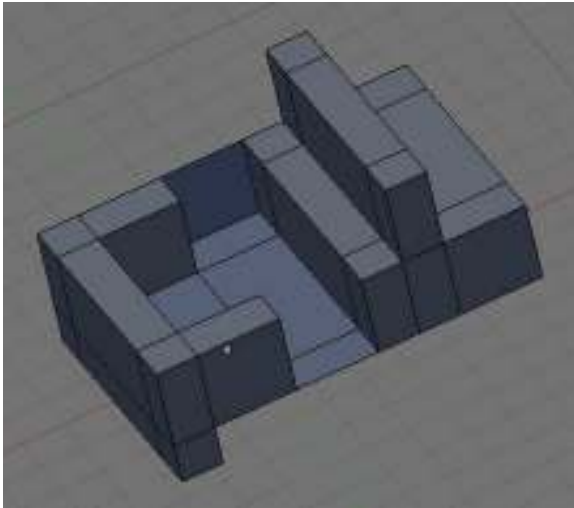


Figure 388

Now in front view (**NUM1**)

Be in **Edge select** mode

Select (**RMB**) the top edge of the door face, and delete (**XKEY**) the "edge". Next, do the same to the corresponding door edge on the other side.

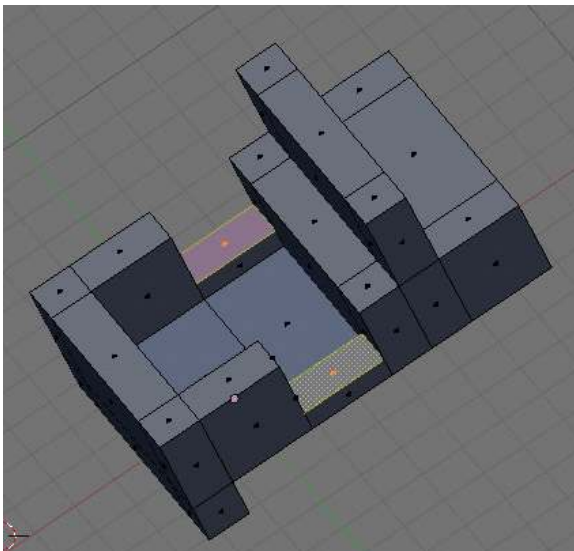


Figure 389

Be in **Face select** mode

Select the faces in the bottom of the door wells, and extrude them up 0.5 along the Z axis.

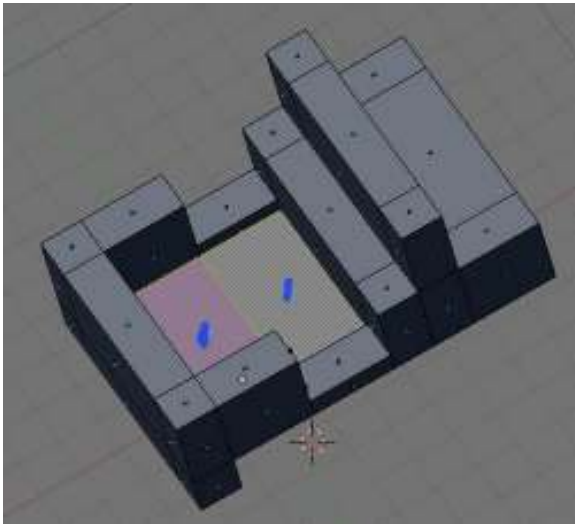


Figure 390

Select the bed panels, and extrude them up .1 along the Z axis (leaving them on top of each other will cause problems with rendering engines down the road)



Figure 391

Noob Note: Be very careful as you extrude the parts up, since they are on top of each other, it's easy to select the face on the outside, rather than the one on the inside. Once you've done the extrusion, check the normal lines by hitting the *draw normals* button in the mesh tools more panel (*if you can't see it, hover and scroll with the MMB*). If you don't get good normals, then hit undo, then reselect the faces, and try again until you get the correct face being extruded (50% chance of getting the wrong face pulled up). Or select the faces with a wrong normal and hit the *Flip Normals* button in the Mesh Tools tab.

Finally, select the whole body and remove doubles, (**AKEY**) to select all then (**WKEY**) to bring up the special's menu. On the special's menu, hit "remove doubles".

Note: removing doubles every couple of steps is a good habit to get into, and will save you time down the road.

Method 2

Make the jeep bed

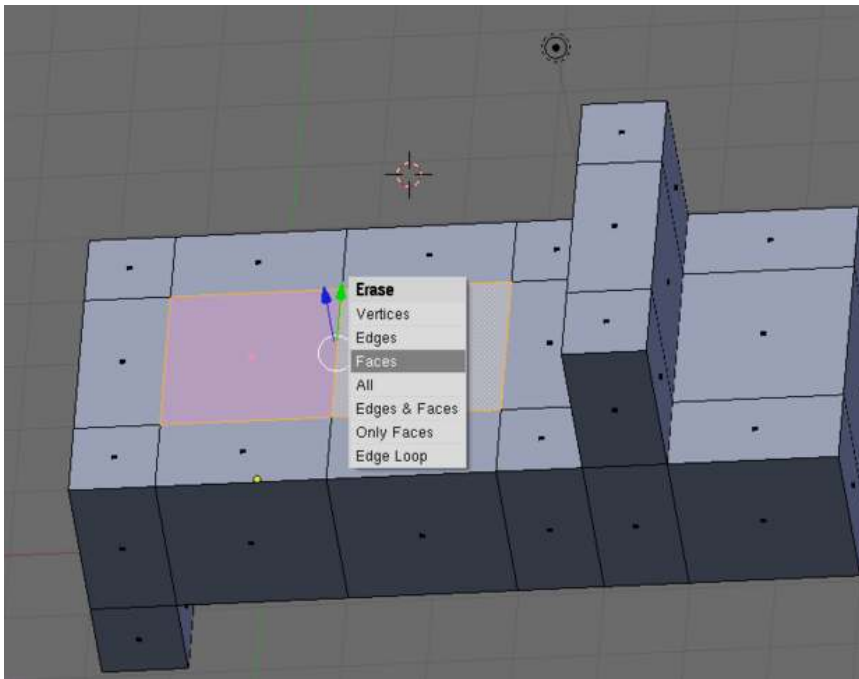


Figure 392 Deleting faces

be in Top view (**NUM 7**), and **Face select** Mode.

Select the two faces in between the bumpers and windshield (**RMB**, then **SHIFT+RMB**).

Now delete both faces (**XKEY**->"faces").

Note: you could also be in edge select mode, and delete (*xkey*) the edge between the two faces, for the same end result.

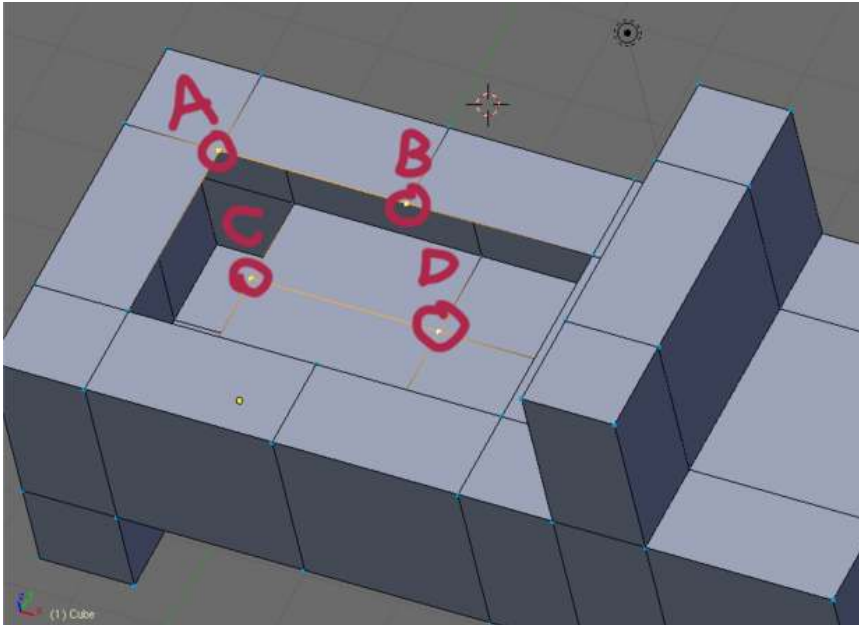


Figure 393 Selecting vertices

You will now add outer faces around the hole we just opened in the model, by selecting two vertices at the top of the model, and two from the bottom of the model.

Be in vertex mode

Deselect all (**AKEY** twice)

Starting at vertex A, circle select (**BKEY** twice) vertex B, C, and D,

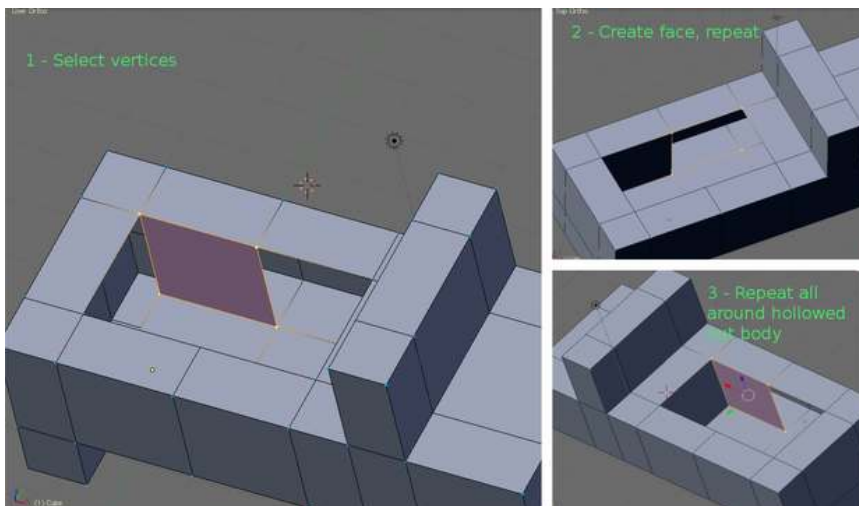


Figure 394 Creating faces

Next create a face (**FKEY**); do this same procedure 5 more times at the vertex groups show in the pictures.

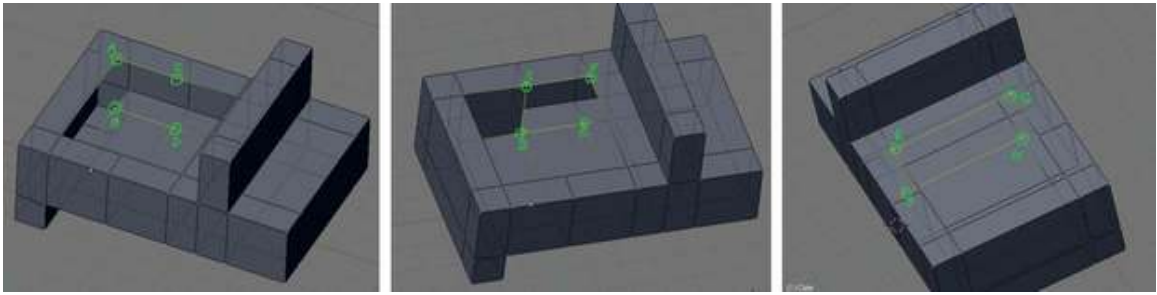


Figure 395

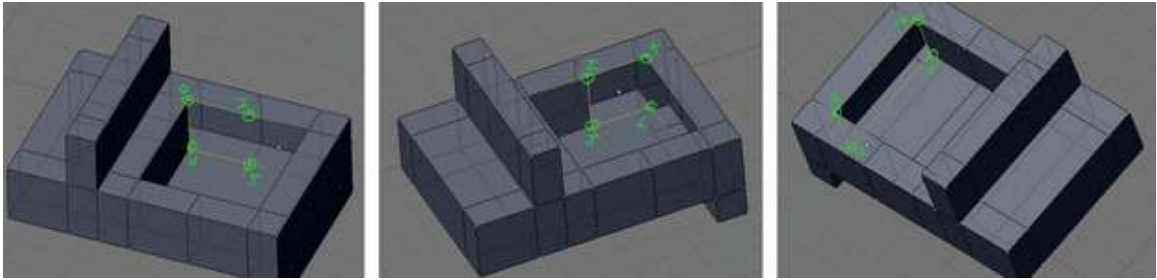


Figure 396

Issues with exposed inside faces (Normals)

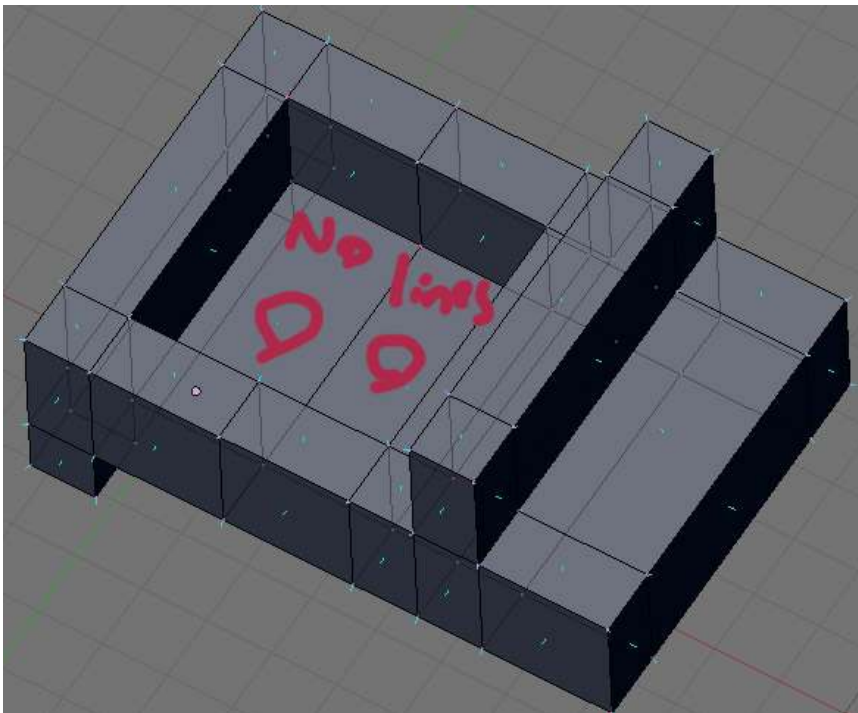


Figure 397

Now that we have gotten the new outer faces in place we can't see into the model any more but we still have a big problem, the floor is actually the outside face of the bottom of the model, this is not good because you should never leave exposed inside faces (the other side of outer faces) on a model.

Exposed inside faces are invisible when looking through them when you apply textures and render it, since nearly all rendering engines completely ignore inside faces (A common problem with many sketchup models, ah the irony!).

You can tell a face is an inside face, by the fact that if you turn on show normals, and show vnormals, on the mesh tools more panel, the outside faces and vertices will have blue lines shooting out of them, while inside faces will not.

Understanding this now will save you lots of wasted hours in the future trying to fix your normals.



Figure 398

Press **NKEY** and turn on the **show vnormals** and **show normals**, and look closely at the exposed interior faces in the floor of the truck bed, they don't have blue lines popping out in the inside, only on the outsides.

Now select textured from the box where you pick wireframe, or shaded, and the exposed inside faces will become invisible when viewed from the inside, but look great when looked at from the outside.

Add the floor of the bed

Lets make a floor for the bed now.

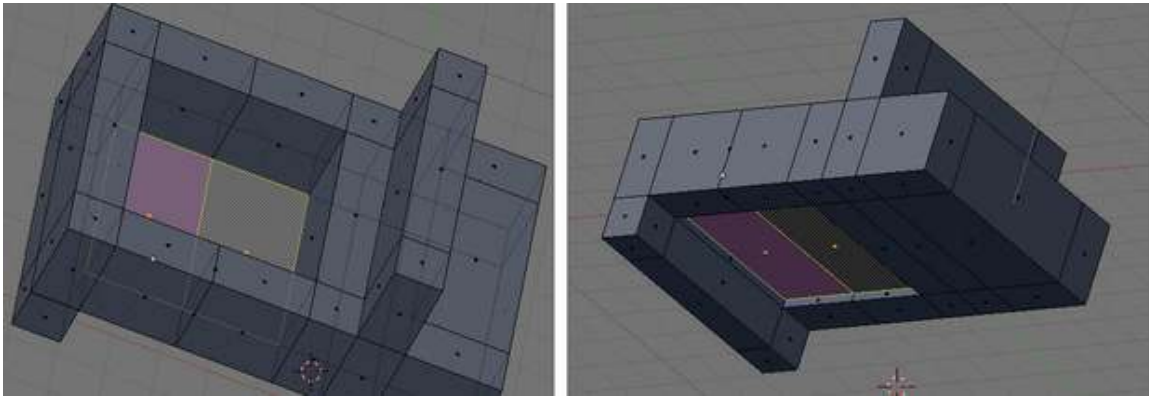


Figure 399

Be in **Face select** mode

Select (**RMB**) both faces of the truck bed,

Extrude (**EKEY**) pick "region" from the popup, and move up along the Z axis (**ZKEY**), by about 0.2.

With the two faces selected, hit mesh -> normals -> flip to make their normals correct for their new position

Now rotate the model till you're looking at it from the bottom, as you can see there is now a hole that needs some extra inside faces removed, and new faces added to complete the bottom.

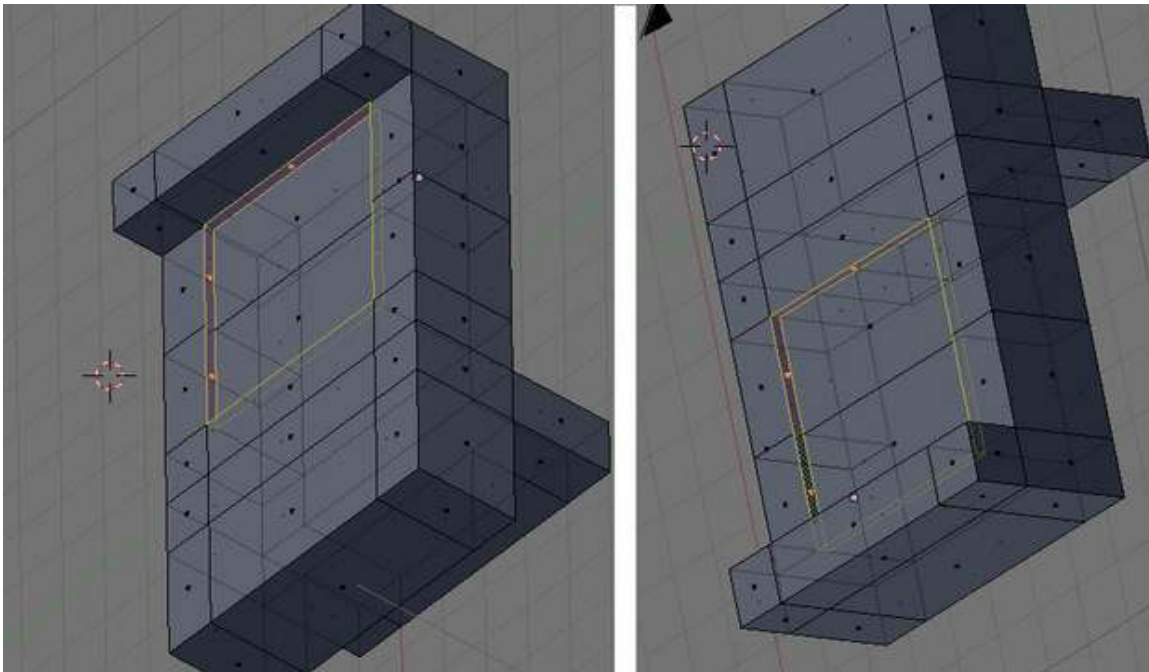


Figure 400

Select each extra face on the sides, and delete (**XKEY**) the "faces" as shown in the photos.

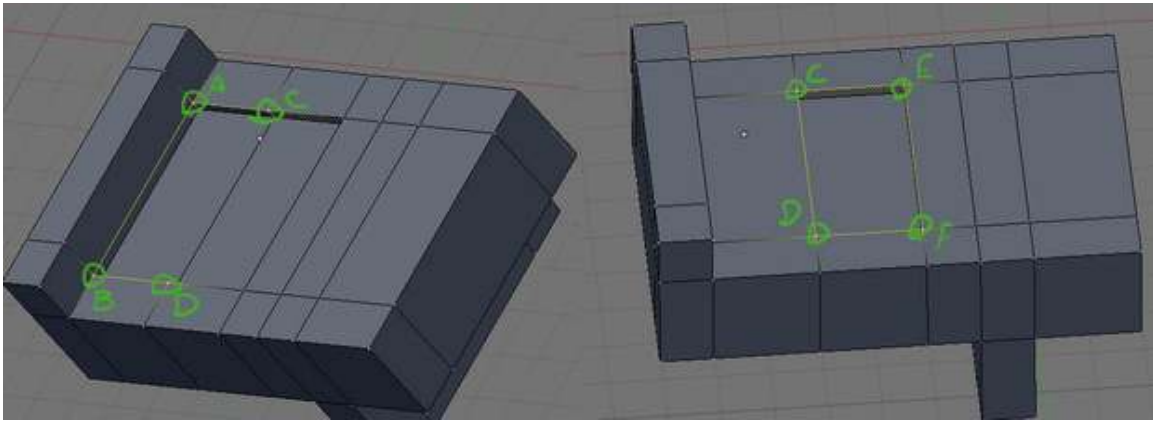


Figure 401

Change to **Vertex select** mode

Select (**RMB**) corners a, b, c, d, and create a face **FKEY**, then do the same for c, d, e, and f.

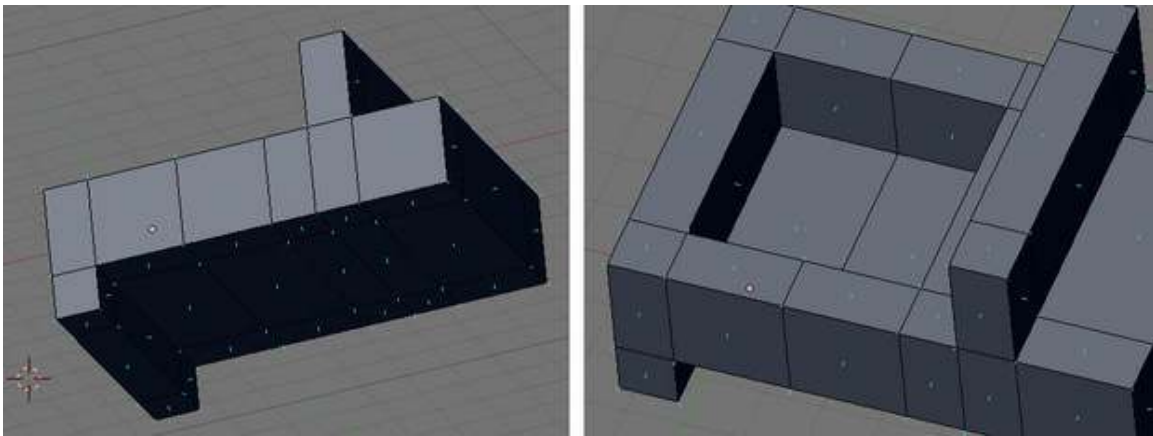


Figure 402

Recheck the normals, and make sure you only have outer faces on the outside of your model.

Make the door holes

Method 2.2 is easier!

Method 2.1 Subdivision

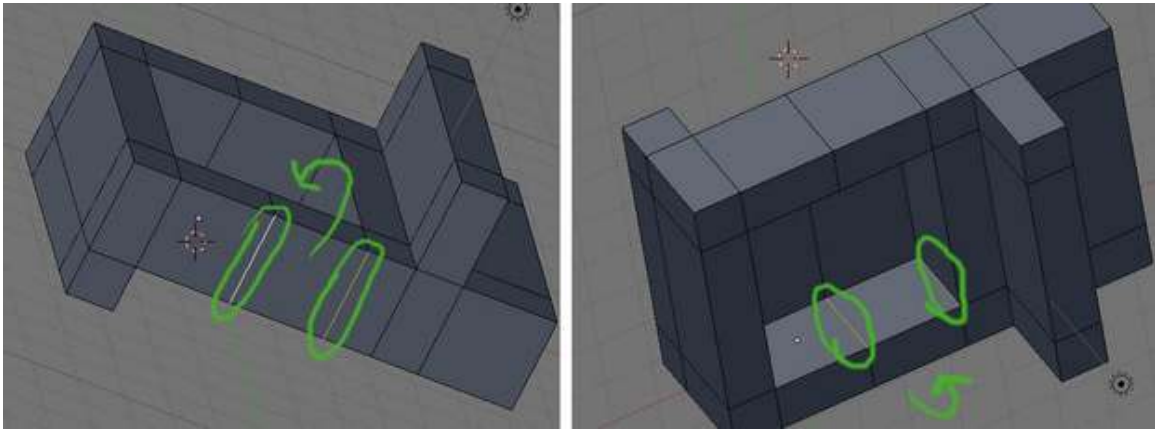


Figure 403

To cut out one door, select the four vertical edges where the door will go.

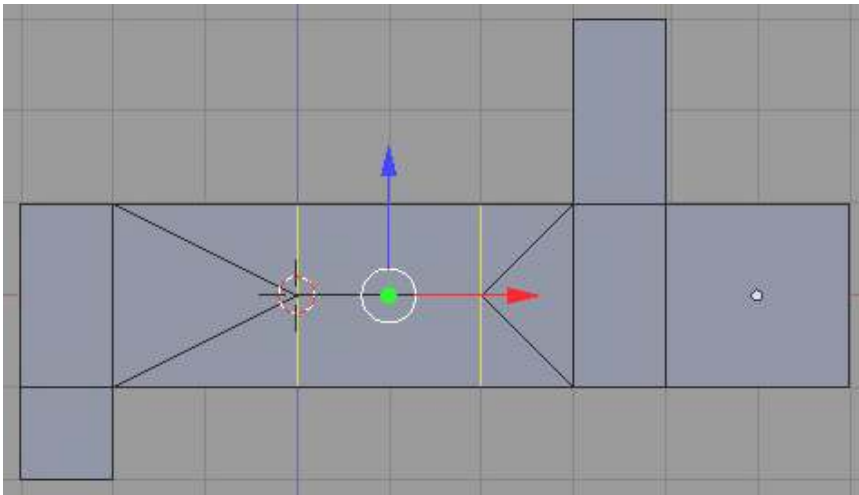


Figure 404 Split the area where the door will go.

Use the subdivide command (**WKEY**) to cut the edges in half. You'll notice that the subdividing will also affect the adjoining faces.

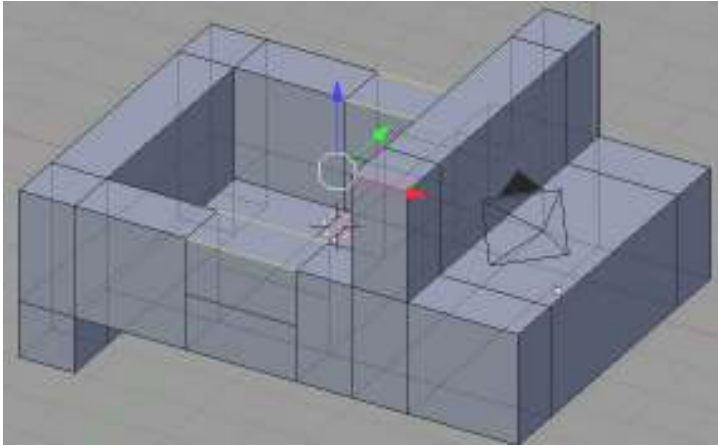


Figure 405 Delete edges for door opening.

Select the two edges at the top of the door panel and delete them, removing the top half of it.

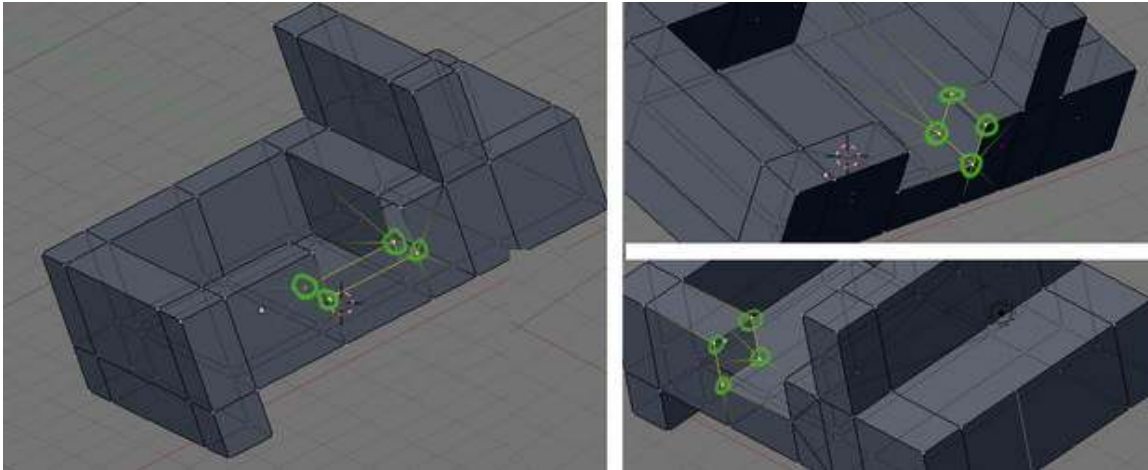


Figure 406

Change to **Vertex select** mode

Now build up faces by selecting the groups of vertex's as shown in the photo, and creating a face (**FKEY**). Do the same thing for the other side, removing/adding edges and creating new faces.

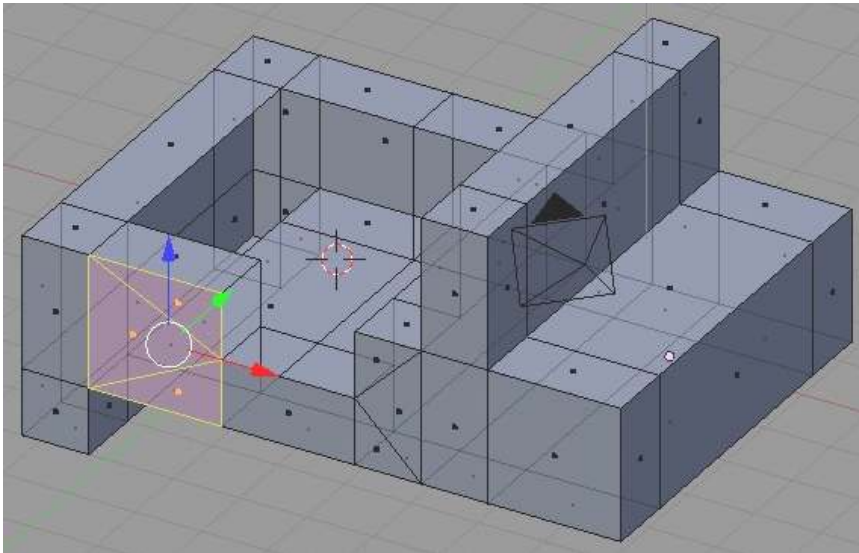


Figure 407 Merging faces.

If you want to clean up the look of the sub divided faces in the model

Change to **Face select** mode, and select the faces that are going to be combined.

Press **FKEY** and a little window will appear titled 'Make Faces'. Click on 'Make FGon' to merge the faces. (As you may have guessed, if an FGon face is created and you want to later undo it, select the 'clear FGon' option in the Make Faces window.)

Note: to merge an FGon to a flat face, select it and hit **CTRL+JKEY**

Pro Note: You don't even have to make an FGon. Just select the door-to-be, Tris and Quads both, and hit **CTRL+J-KEY**.

Noob Note: Q. What is an FGon??

A. It is a fake polygon, a way to hide triangles and quads on flat faces.

Method 2.2 Multicut

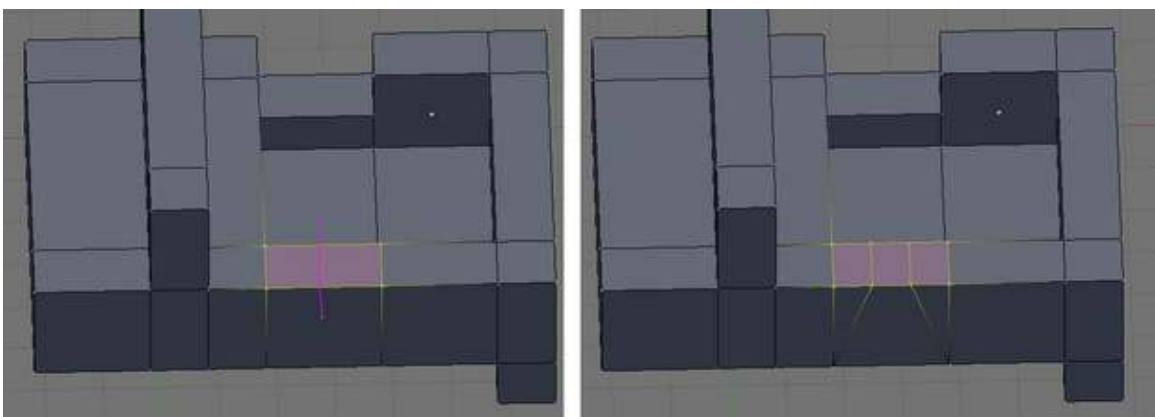


Figure 408

Be in frontview (NUM1).

Select the top edges where you want the doors

Multicut (KKEY), enter two in the popup, draw a line along the axis of the cut, as shown in the photo, and hit **ENTER**.

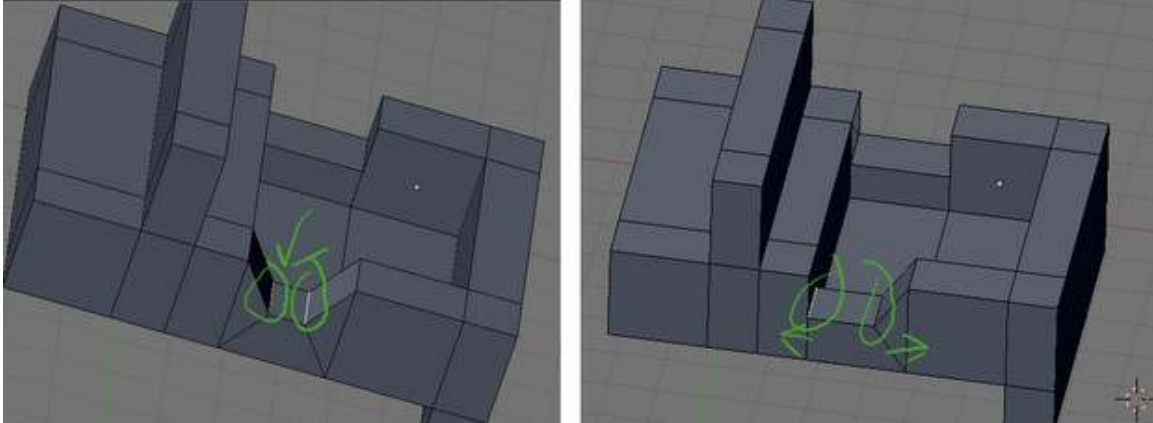


Figure 409

Then move the newly created edges down and towards the sides.

method 2.3 (Noobie)

Select top face of where door is going to be

FKEY to extrude, pick "Region" from the popup menu
ZKEY to lock in Z direction
 hold **CTRL** to snap to whole BUs and move down one BU

Select 2 top edges that look like they have no widths

XKEY to erase, pick "Edges" from the popup menu

Two faces will be missing. For each individual missing face, Select top and bottom edges

FKEY to fill in the missing face.

58.3.4. Resizing the bed and windshield

Go ahead and resize the widths of the bed and windshield.

As a precaution, Remove doubles before starting!

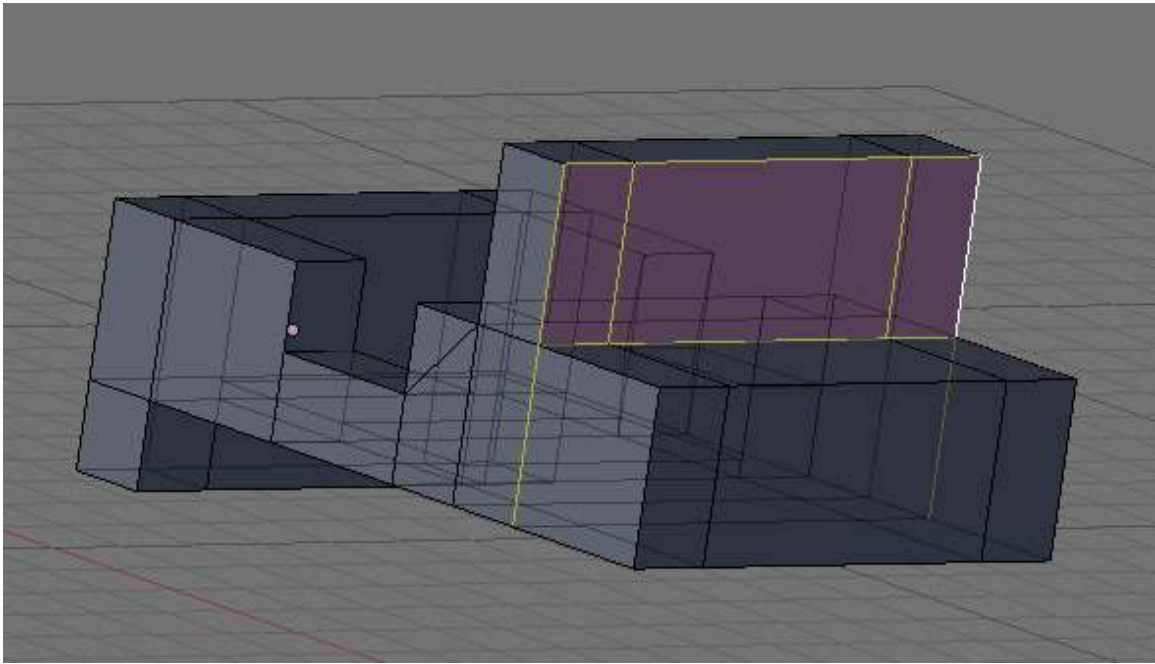


Figure 410

To narrow the width of the window, change to **Edge mode** and select the edges shown in the picture, then move (**GKEY**) along the X axis.

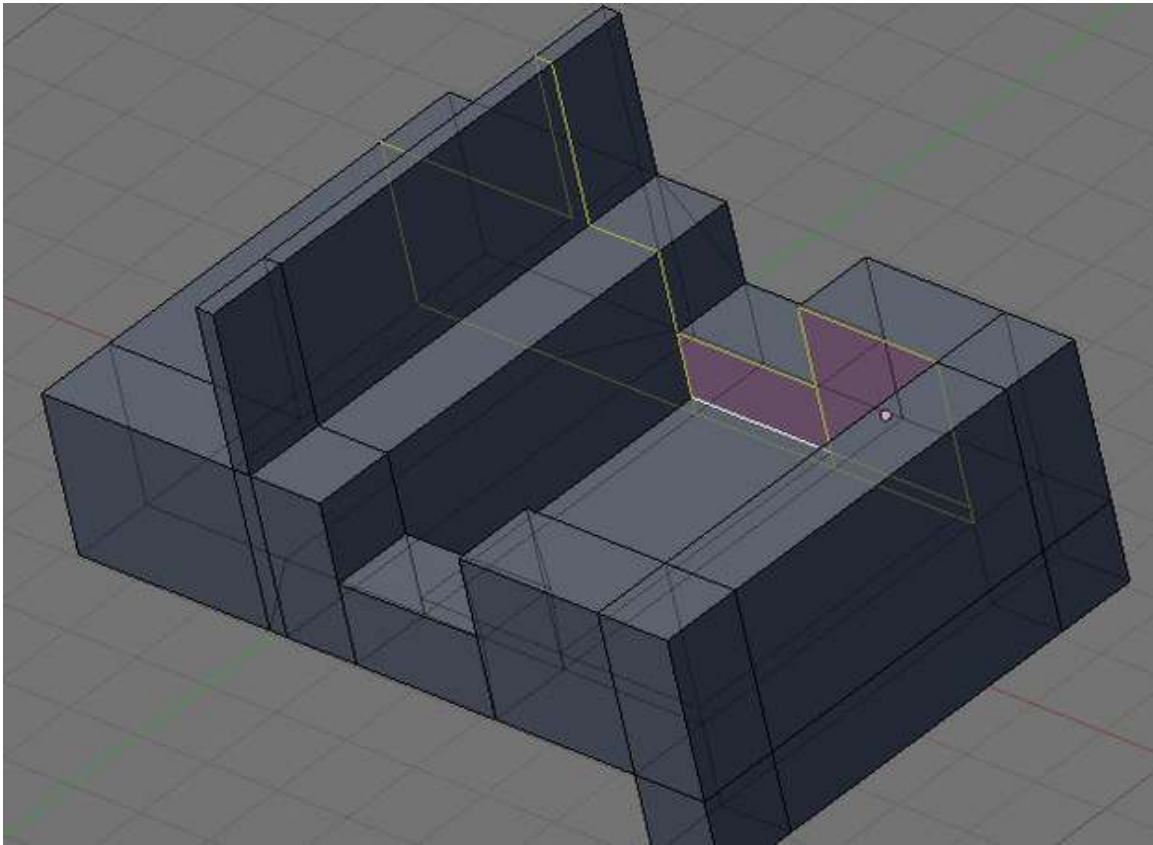


Figure 411

To narrow the width of the side, change to **Edge mode** and select the edges shown in the picture, then move (**GKEY**) along the Y axis.

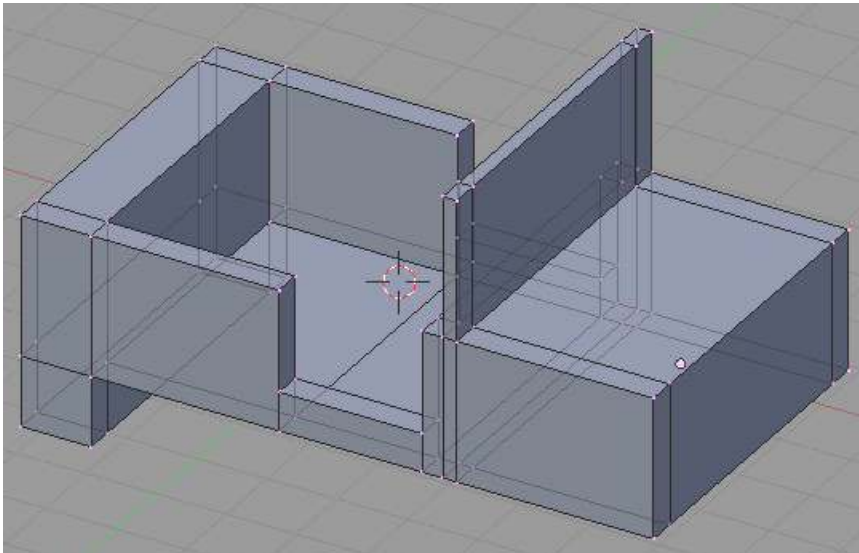


Figure 412 Adjusting the widths.

Always move as many vertices or edges at the same time to not only work faster but to make sure they are moved equally. The use of circle select, loop select, and **SHIFT** while moving vertices is very helpful in fine movements.

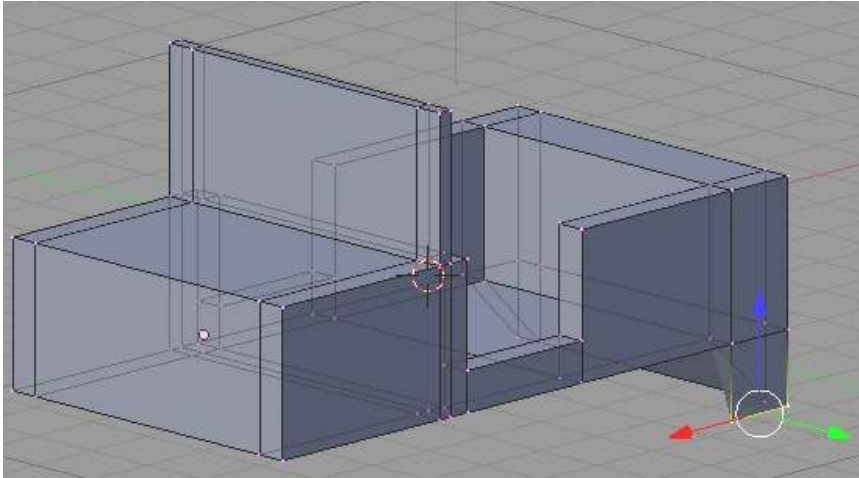


Figure 413 Bringing the back to a point.

If you want an object to come to a point such as a wedge from a cube, merge vertices.

In this example, the lower back area will be modified. Select two vertices to join together and press **ALT MKEY**. Select the option for your merging. 'At First' or 'At Last' will probably be the option that will work here.

Play around to see what each merge option does. After the merging, Blender will tell you how many vertices were removed.

Be certain to "remove doubles" (**WKEY**), as merging creates lots of em. (*you can also find it on the menu Mesh ==> Vertices ==> Remove doubles*)

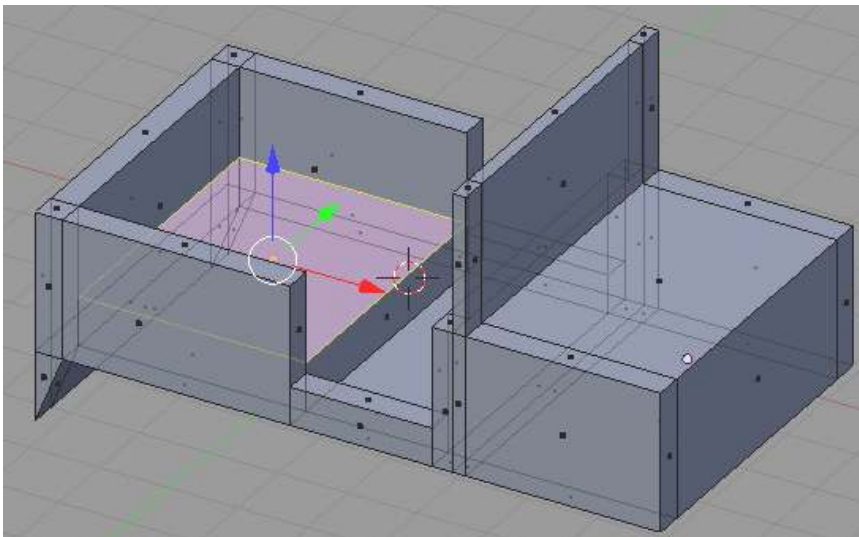


Figure 414 Raise the flatbed.

Extrude the bed surface upward. This is only useful in hiding the tires that we'll add later in the tutorial. Alternatively, you could make two boxes to hide them.

58.4. A Touch of Detail

58.4.1. Adding an Engine Hood

Method 1

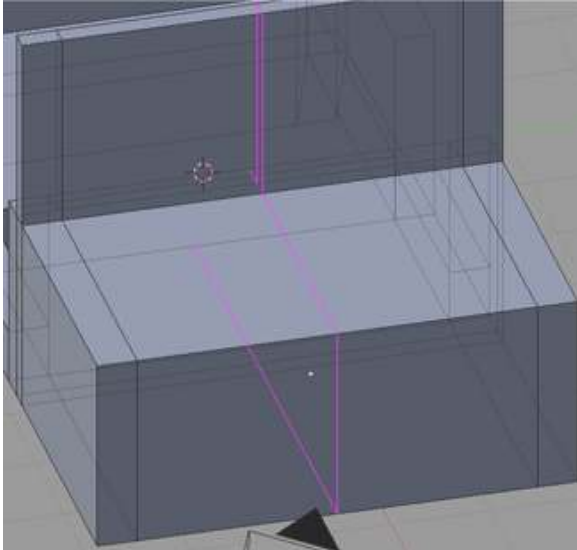


Figure 415 Loop Subdivide select.

Let's add some detail to the model - how about the hood? First thing to do is add some edges to the front of the jeep. Press **CTRL RKEY** to enter Loop Subdivide. A pink loop will appear around the mesh. Put the cursor over the area to get the example picture to appear.

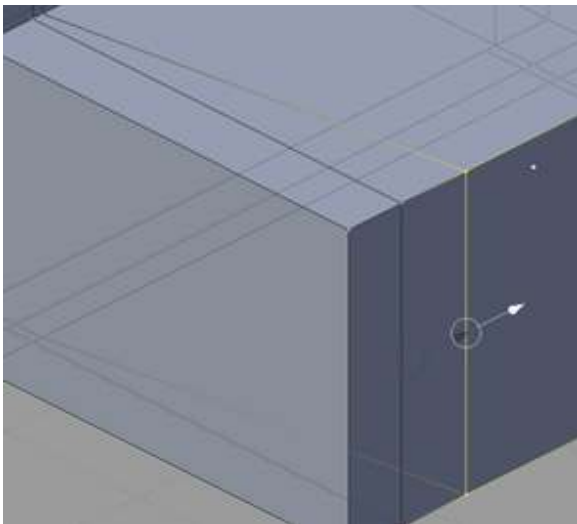


Figure 416 Aligning the hood.

When the loop is in the right place, **LMB** click. The place to put the actual cut can now be selected. Do this twice - once for each side. Move and align the resulting edges to form an angle to the front and bring the window vertices in.

Extrude the hood surface up a small amount. We don't want it too high, just high enough to catch the light.

Zoom in and select the top-front hood edge created from the extrusion. Drag it out along the X axis. Select the now diagonal face of the hood extrusion. Extrude from it. The result will come out of the surface at a diagonal angle. Take the resulting vertices and move them close to the front of the jeep.

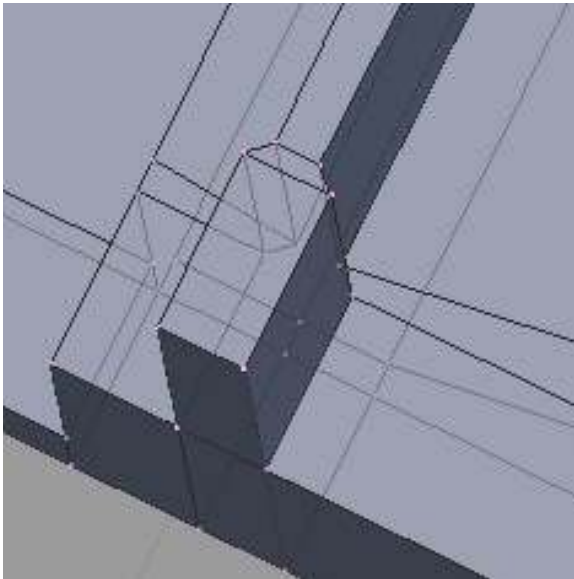


Figure 417 Formatting the window.

Using the additional lines from the loop subdivide you can improve the shape of the window

Method 2

Another way to raise the Jeep hood. It doesn't require loop cuts, so is a bit simpler.

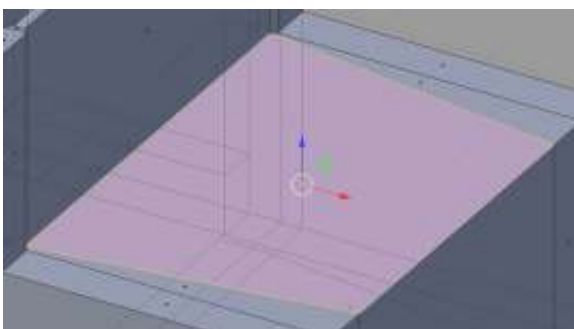


Figure 418 Hood extrusion.

Step 1. Make Inner Square.

Switch to Top-view NUM 7, or slightly rotated off for easier viewing. Face-select the top square of the hood.

Do **E**-xtrude, then hit <esc>. NOTE: this WILL make a new surface, hitting <esc> doesn't cancel the extrude, just makes its location to be exactly on the old surface.

Do **S**-cale and type 0.9 and hit <return>. Now you will see the new surface as a smaller square (or really rectangle) on top of the jeep hood square.

Noob Note: What I had to do is **S**-cale on **Z** and type 0.9, otherwise the New Surface For the hood would end up shooting through the Windshield.

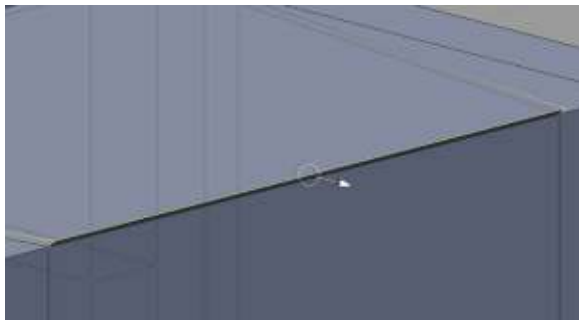


Figure 419 Bring out the top front edge.

Step 2. Shape Hood.

Still in Top-view NUM 7, deselect all.

Box-select the right two vertices of the new square, towards the front of the jeep.

Do **S**-cale on Y, and type 0.8 and hit <return>. Leave the two vertices still selected.

Do **G**-rab on X, and type 0.2 and hit <return>.

Face-select the resulting quadrilateral.

Switch to Side-view NUM 3.

Do **E**-xtrude on Z and type 0.2 and hit <return>. I used 0.2 to exaggerate the screenshot a bit, you probably want 0.1 instead.

Now you should have a raised hood on the front of the jeep.

method 3

Another way to extruding the Hood of the jeep including a lip that comes over the front.

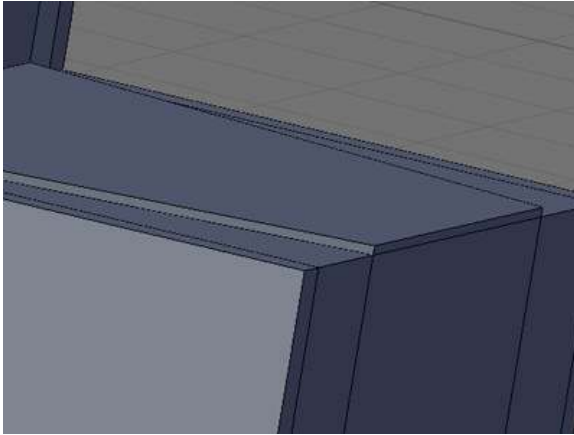


Figure 420

Step 1: In face select mode (**CTRL+TAB-faces**), select the top of the hood and press **EKEY** to extrude it. Only extrude it a small amount.

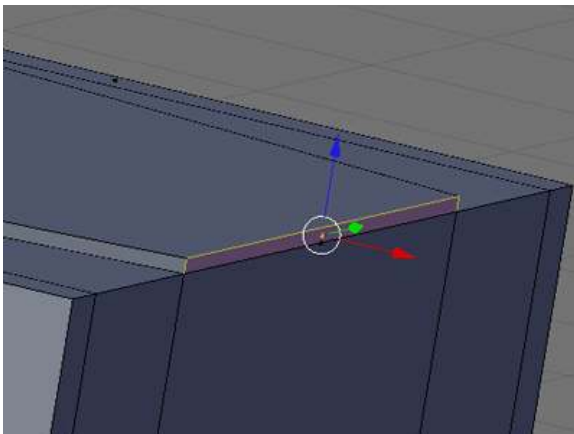


Figure 421

Step 2: Staying in Face select mode, here is the tricky part. Select the small face on the front of the hood you just extruded. Then, hit **EKEY** then **ESC**. This creates a duplicated face on top of the one you selected. Do not click or move the mouse between these two keystrokes.

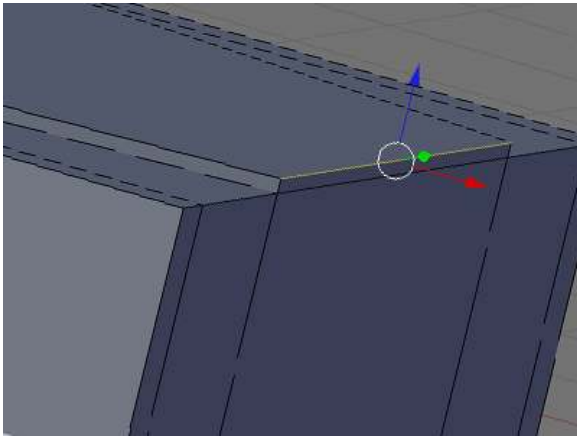


Figure 422

Step 3: Now deselect the selected face by hitting **AKEY**. Then enter Edge select mode (**CTRL+TAB-edges**) and select the top edge of the face you just deselected.

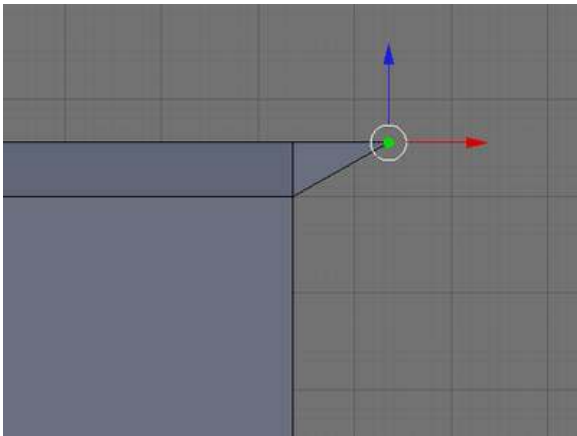


Figure 423

Step 4: Hit **NUM 1** to go to the side view. Now using the red X arrow pull the edge out a little further than you pulled up the hood itself then hit **LMB**.

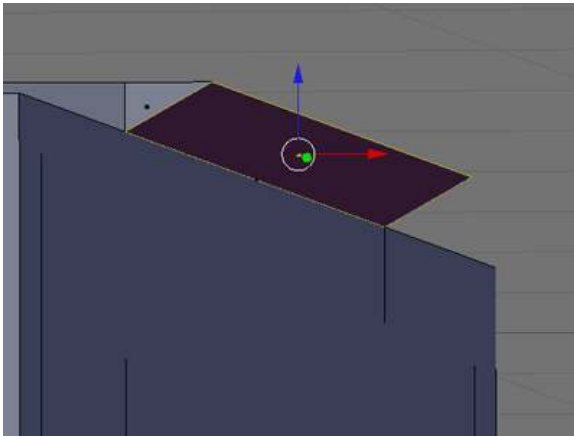


Figure 424

Step 5: Now rotate just enough so you can see under the wedge you just made. Go back into Face select mode (**CTRL+TAB**-faces) and select the face on the underside of it.

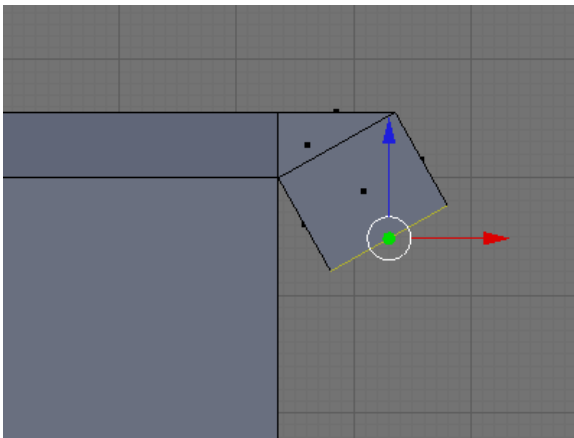


Figure 425

Step 6: Go back into side view with **NUM 1**. Now hit **EKEY** to extrude the face a little with the mouse. When it's a good size hit **LMB**.

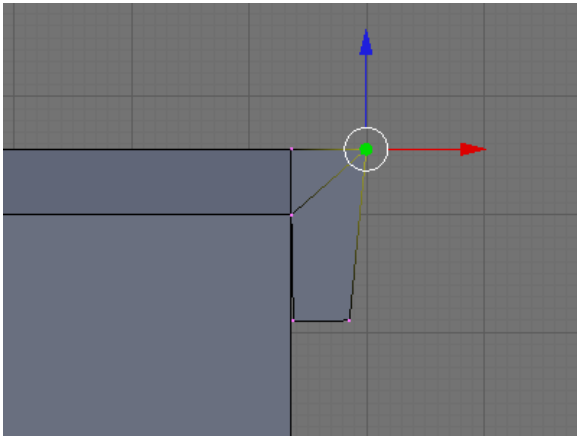


Figure 426

Step 7: Now make sure the "Select only visible" button is turned off ("Occlude background geometry" in later versions) and go into Vertex select mode (**CTRL+TAB**-vertices). Play around with the vertices pulling them a little closer to the front of the jeep. It's best to select using the box select (**BKEY**) or the Lasso (**CTRL+LMB**).

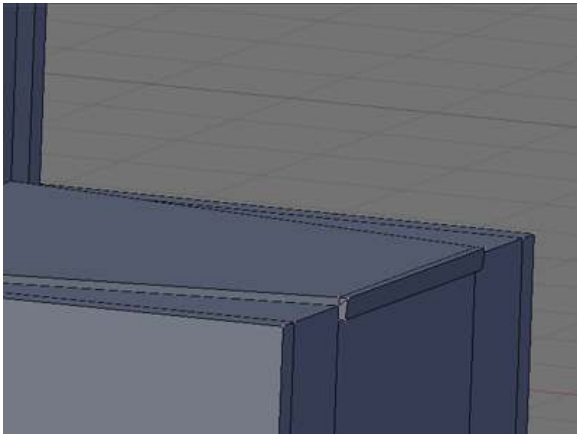


Figure 427

The end result should be an extruded hood with a lip.

58.4.2. Add a Fender

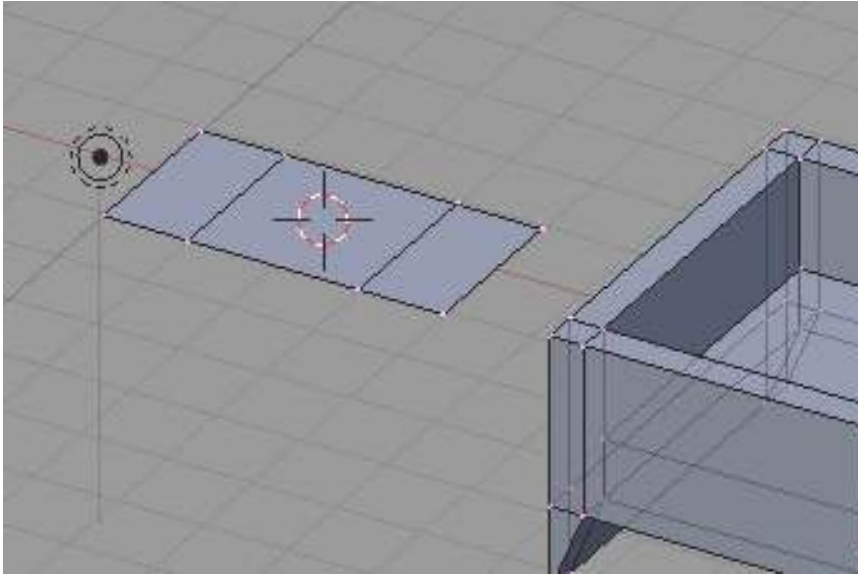


Figure 428 Starting the fender.

In the topdown view (**NUM 7**), add a plane. Extrude an edge twice to result in three connected planes.

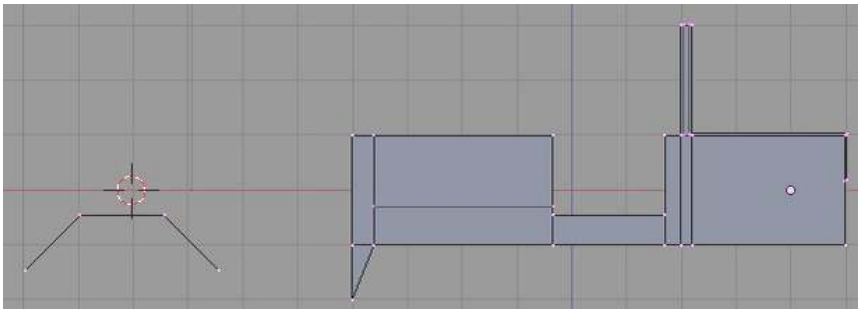


Figure 429 Bend the fender.

Pull the sides down to form a trapezoid shape and reduce the width.

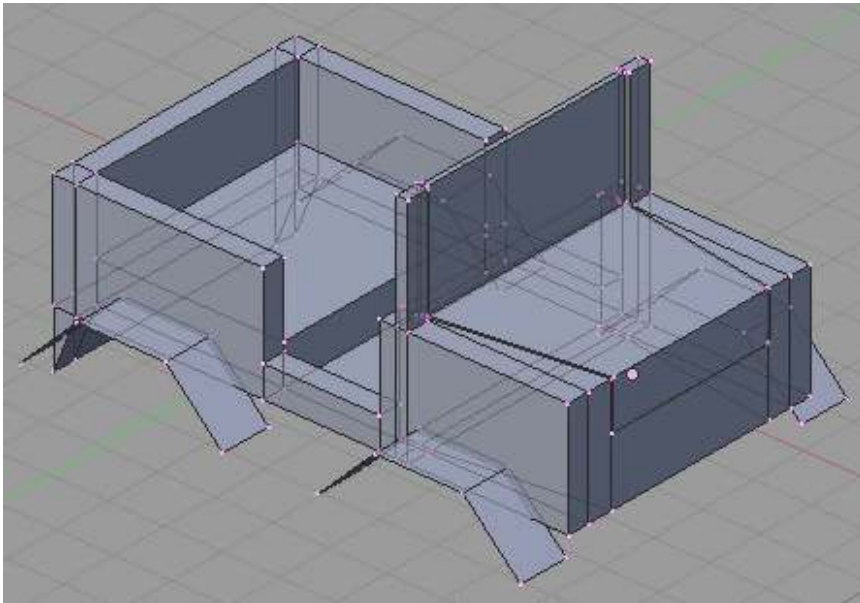


Figure 430 Model with fenders.

Once it is in the desired shape, select the three faces and duplicate it. Press **SHIFT DKEY** and all selected vertices, edges, and/or faces will be duplicated. The copy will automatically be grabbed for moving.

Move the duplicate to the jeep body and repeat the duplication two more times for a total of four fenders.

Noob note: A good idea is to first position one fender, then copy it and restrict movement to x or y-axis. Then copy both fenders and move the two new copies along x or y-axis. Much simpler than trying to position four fenders individually.

58.4.3. Add a rocket launcher mount

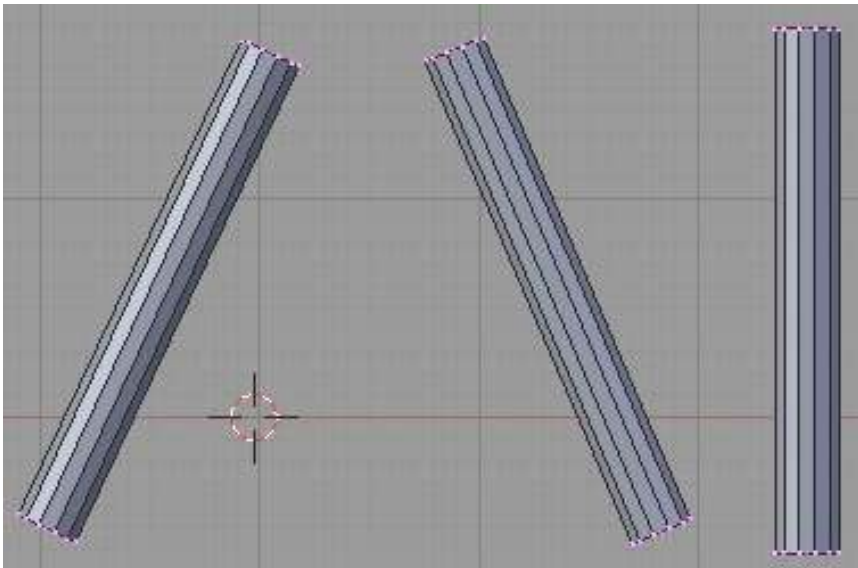


Figure 431 Setting up the tripod in **NUM 1** view.

We'll move on to making a tripod support for the rocket launcher.

Add a cylinder mesh with 12 vertices then scale and size it so that it looks like a tube. Once you have it to a size you like,

Duplicate it twice for a total of 3 cylinders.

Rotate two of the cylinders in the **NUM 1** view by **LMB** clicking on the white circle that appears when the cylinder is selected in rotate mode.

The picture on the left is an example of the end result.

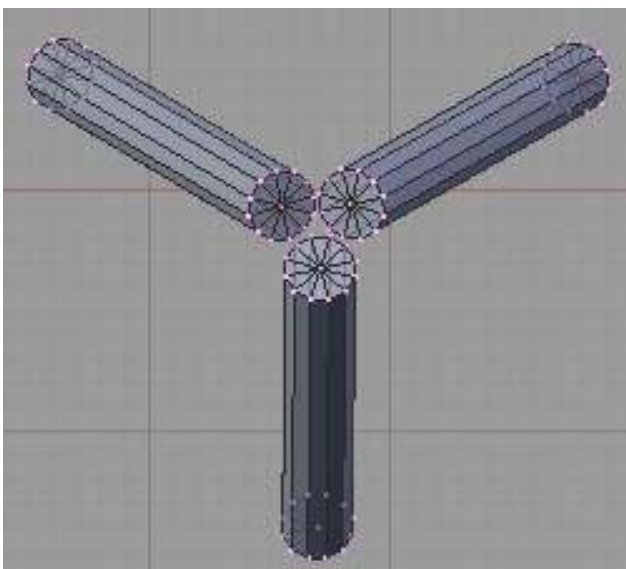


Figure 432 Setting up the tripod in **NUM 7** view.

Change to overhead view (**NUM 7**) and put together the three cylinders so the tops come close together. Now all three can be selected and moved or rotated accordingly.

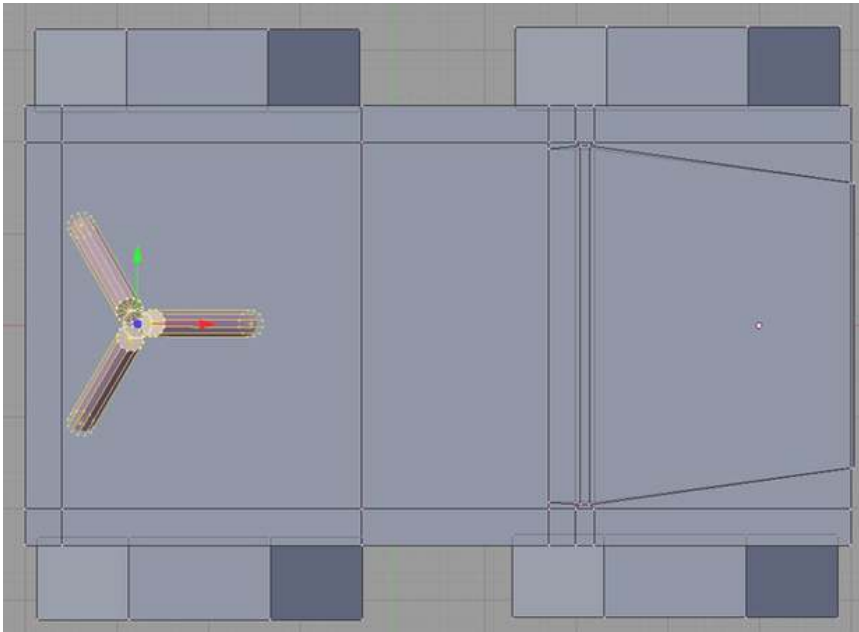


Figure 433 Putting in the tripod.

Move the tripod onto the jeep flat bed. The final steps are to select your materials and rename the object (described in the wheel section). This will complete our simple jeep model.

58.5. Subsurf

Since we expect a jeep body have squared up edges, we won't subsurf this part of the model.

58.6. Optional Activities

Feel free to add anything you see fit such as bumpers, guard rails, doors, steering wheel, lights, etc. You can either have them on the same object or separate objects (useful if you want to move them around).

59. Simple Vehicle: Some Assembly Required

59.1. Techniques

You should know how to:

- Do everything discussed in previous tutorials

This section will recap and introduce:

- Append a file
- Duplicate an object

59.2. Overview

The objects for the simple vehicle have been made if you have followed all the previous Simple Vehicle tutorials. Putting it all together will come very easy now.

59.3. Appending the File

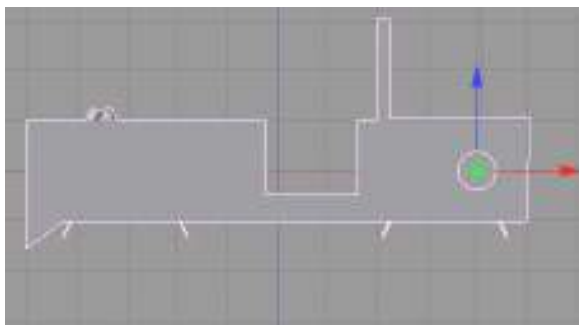


Figure 434

If you have the jeep body file open, keep it open. Otherwise, open the file for the jeep body.

In Object Mode, go up to **File > Append** (*Append or Link* in later versions), about 3/4 of the way down the menu. The Find file window will appear. Go to the location where the jeep seat was saved. When the .blend file is clicked, you'll go into it as if it is a directory.

Here we have the categories of Camera, Lamp, Material, Mesh, Object, Scene, Text, and World. We are interested in the seat object, so click on Object. Now there are three items: Camera, Lamp, and Seat. That is, it will say Seat if you named your object Seat. This is why it is useful to rename your

objects, materials, etc. If you forgot to rename the object, it will be called Cube (default for our starting mesh).

Noob note: You must be sure you're appending, and not linking! If you try to duplicate it, and you get an error, then it's probably linked, there will also be "li" to the right of the ME: object name button. Ways to be certain you're appending are: when selecting the file, look at the bottom for the append/link options and make sure "Append" is selected; In the file menu, don't choose "Append or Link (Image Browser)", you need to use the option above it.

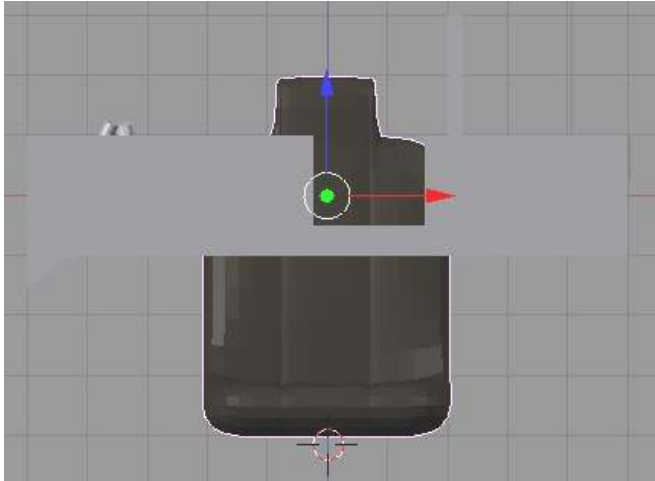


Figure 435

What should happen after selecting Seat and the button 'Load Library' is the seat will pop into our file where the 3D cursor was.

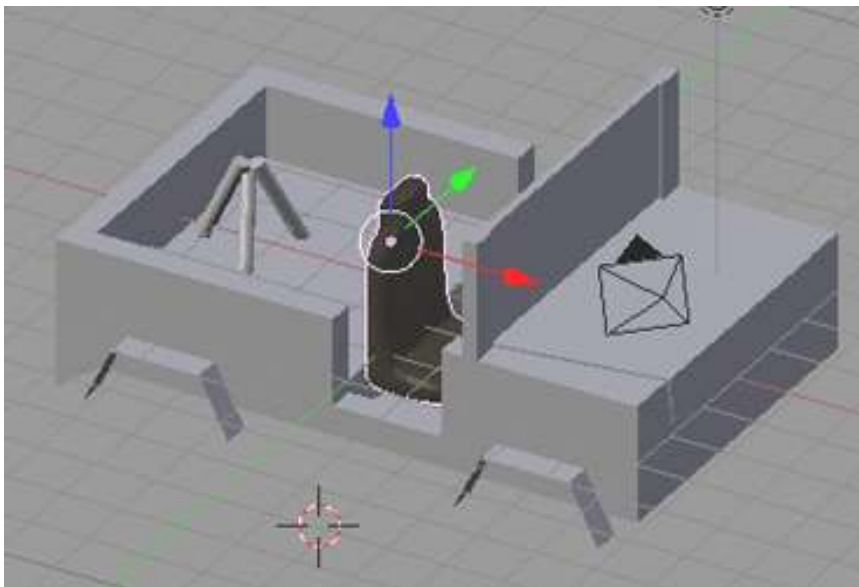


Figure 436

It will definitely need to be scaled, rotated, and/or moved to the right position. One way to do this is to rotate the seat about the Z axis -90 degrees by pressing **RKEY ZKEY NUM9 NUM0 NUM-**.

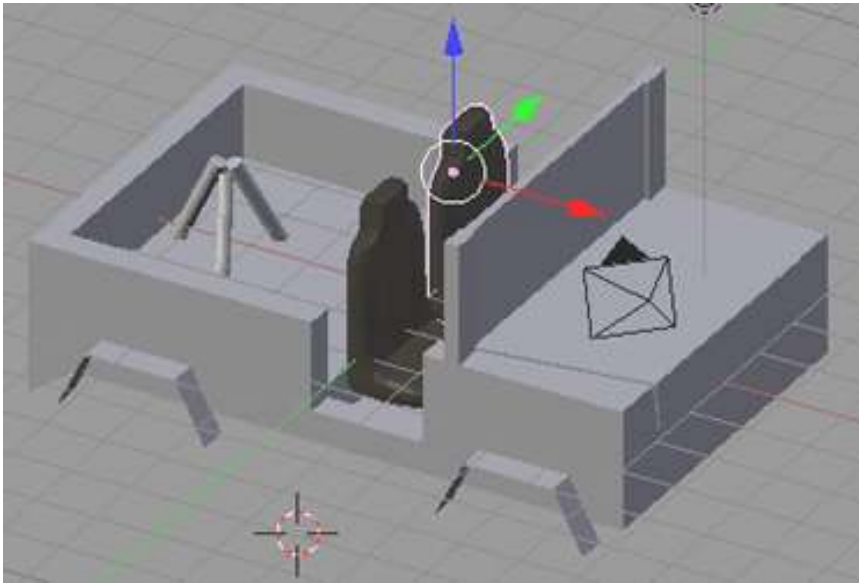


Figure 437

After placing the seat in the jeep body, let's make another so we have a seat for the driver and passenger. Still in Object Mode with the seat selected, duplicate it (**SHIFT DKEY**) and slide it over. After duplicating it, it will automatically go into grab mode. If you **RMB** or **ESC**, it will still be duplicated - just sitting on top of the original.

Noob note: Duplicate appended objects in object mode, not edit mode, as they are separate objects and do not share the same edit mode space (each appended object has its own edit world, unless you join them together)

59.4. Rinse and Repeat

Append the file again to place the wheel object and rocket launcher in the file. Scale, rotate, move, and duplicate each object accordingly. Depending on the position of your camera, you may or may not have to make all four tires. Remember that the only important parts to draw are those that will be seen!

Set the ball over the tripod. The fun part is rotating the rocket launcher since the center of it has been moved to the ball joint.

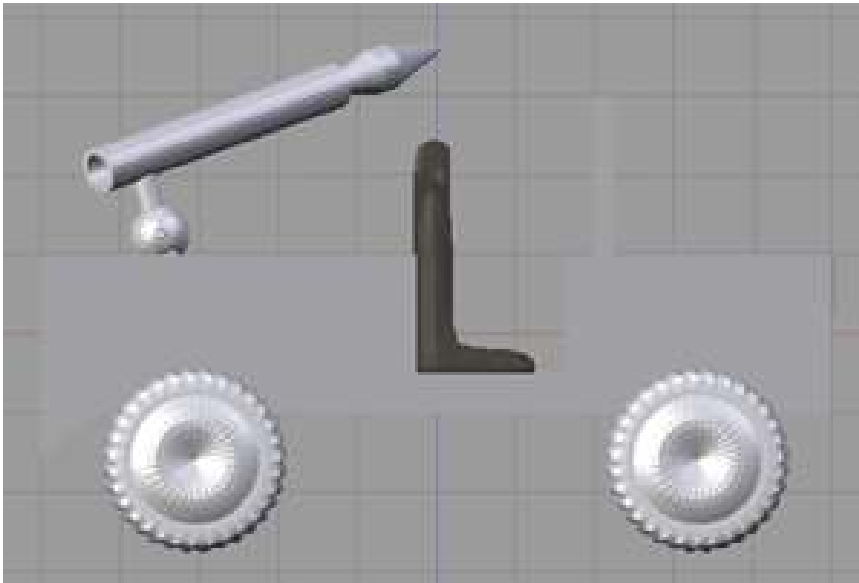


Figure 438 All objects

59.5. Parenting

Be sure to parent each item to the jeep chassis, by selecting the item, lets say the tire, then also selecting the chasis (**RMB** then **SHIFT+RMB**) and then **CTRL+PKEY**, and select "make parent" from the popup menu.

59.6. Final things



Figure 439 Rendered jeep

The last thing is to apply materials to your objects!

You can apply material in either object, or edit mode. As you might have noticed when you select something in object mode the entire object is selected, in edit mode on the other hand individual faces can be selected, and painted.

59.6.1. Make the windshield look like glass

You have to create a new material and assign it to the appropriate faces.

In edit mode, select the faces you want (make sure you select both sides of the windshield including: front face, back face, and top face).

On the Editing section F9, under Links and Materials, click the 'New' button under Materials then click the 'Assign' button. This assigns the new material to the selected faces.

You can then go into the Shading section F5 and adjust the material to be glass, by setting the alpha to .20, and hitting the ray transparency button. Three more ways of making a material that looks more like glass are shown in Material Glass¹.

Make sure you are editing the correct material (if you have just the one texture on the jeep, the default name for the new material should be something like "2 Mat 2").

¹ <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FEvery%20Material%20Known%20to%20Man%2FGlass>

Noob note: Glass is only transparent when you render it. in order to render it you will need to position a few lamps around the jeep, and move the camera around so that it can see the jeep (view->camera).

59.7. Extra



Figure 440

Other possible ideas



Figure 441

59.8. Additional Tutorials

Append and Link Video Tutorial: <http://www.youtube.com/watch?v=69ZBIDrOIIY>

60. Modeling a 3D Parachute in Blender

Blender3d Aligned to view issue

This tutorial relies on objects being created so that they are aligned to the view that you're looking through. Versions 2.48 and above have changed the way this works. Visit Aligned (http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Aligned_to_view_issue) to view issue to understand the settings that need to be changed.

60.1. Remove the default cube

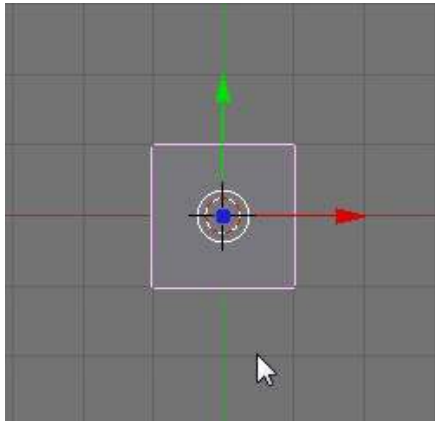


Figure 442

When you first open blender you will see the default Blender user interface (UI) layout and the default cube.

Delete the cube by selecting it with the Right-Mouse button and then clicking the X key.

60.2. Add a cylinder

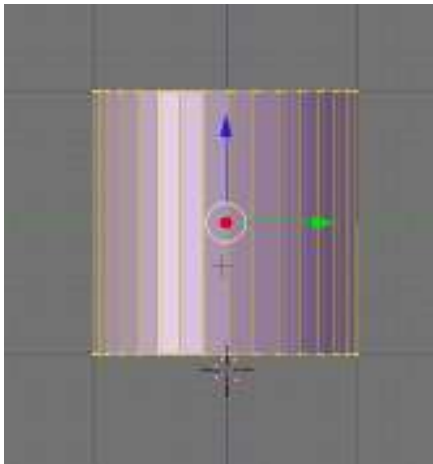


Figure 443

Be in **Side** view.

Add a Cylinder by clicking Space-bar> Add> Mesh> Cylinder. This will bring up a small window in which you can set several parameters including vertices, radius, Depth, and cap ends. We will leave the settings in their default state. Press OK to add the cylinder to the scene.

In earlier versions, Blender would automatically take you to **edit mode**, otherwise you should switch to **edit mode** by pressing the **TAB** key.

60.3. Remove the bottom row of vertices

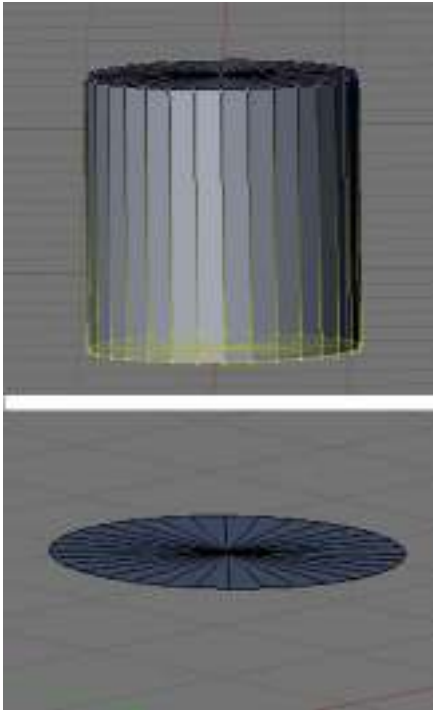


Figure 444

Next you will want to delete the bottom row of vertices. To accomplish this go into side view by pressing “3” on the num-pad.

Ensure that you are in **Vertex select** mode by pressing Ctrl-Tab and selecting “vertices”.

Press **AKEY** twice to deselect everything

Press the B key, and box select the bottom row of vertices, and press the X key to delete and then confirm.

60.4. Extrude and scale to shape

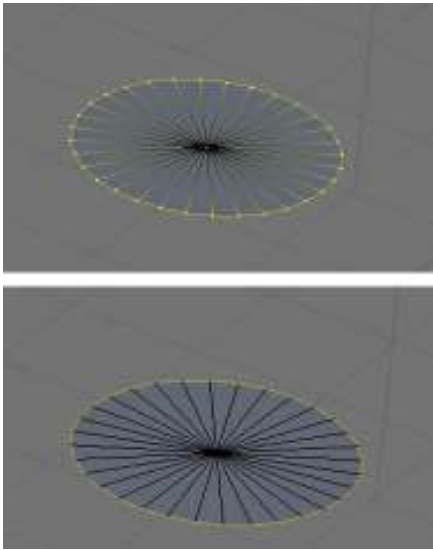


Figure 445

Be in **Vertex mode**

Select all the vertices around the outside of the circle (you can do this with circle select **BKEY** twice). The center vertex must not be selected.

Switch to **Edge select** mode by pressing **CTRL+TAB** and then Edges.



Figure 446

Press the **EKEY** key to extrude ("region") the selection and the **ZKEY** key to constrain to the Z axis.

Drag the edges a small ways down then click the left mouse button to release.

As this makes a right-angle at the sides and parachutes to don't have straight edges, we need to scale the selection outward.

Press the **SKEY** key and move the mouse away from the model, you will see that the edges get smaller and bigger.

Scale them out a small ways then left click to release. You will want to practice a little with the scale amounts till you can make a realistic parachute shape.

Noob note: if you're seeing the popup with regions, edges, etc., when you extrude, then you're not in **Edge select** mode

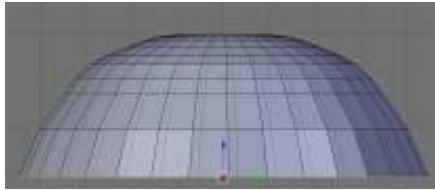


Figure 447

Continue extruding and scaling till you have a shape like the one shown on the right.

When you're done making a nice shape, be sure to select all (**AKEY** twice), then hit **WKEY** and pick "remove doubles"

60.5. Make the top more rounded

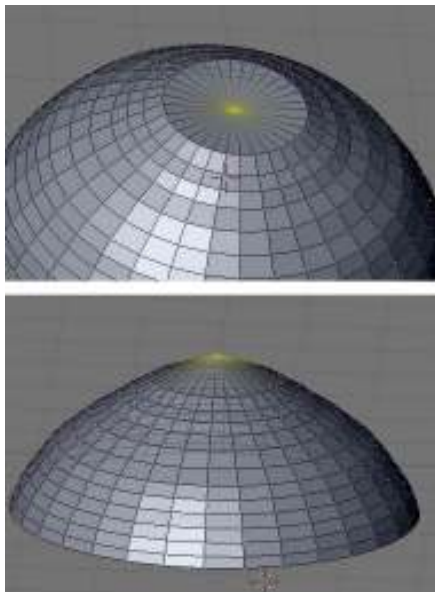


Figure 448

Be in **Vertex mode**

Select the center vertex in the middle of the original top half of the cylinder, use circle select to get it, otherwise you will be **RMB** a number of times to find it, and move (**GKEY**) it up long the Z axis (**ZKEY**).

Select all **LKEY**, and then hit **WKEY** and remove duplicates, in case any were created.

60.6. Extrude the parachute straps

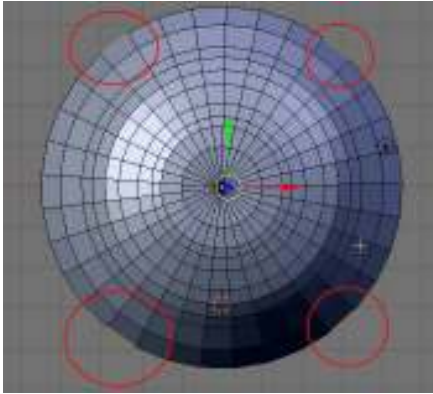


Figure 449

Now go into top-view by pressing the 7 key on the num-pad. Here you will select ~4 edges at opposite sides of each other.

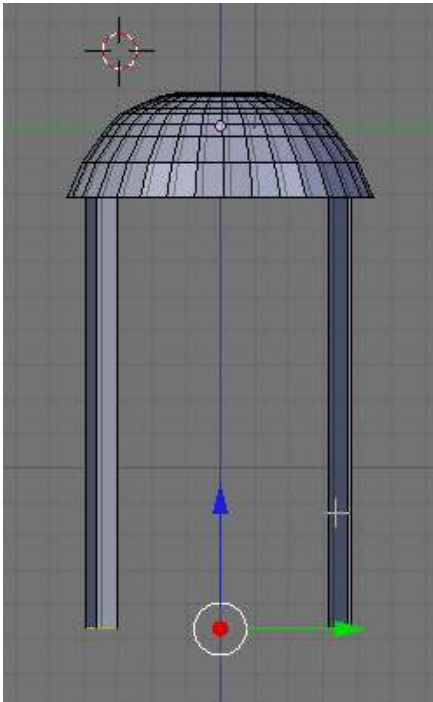


Figure 450

Go back into side-view (num3) and extrude downward by pressing the E key and then the Z key.

60.7. Merge it together



Figure 451

All that remains to finish your parachute is to press Alt-M then “At Center” which will merge all selected vertices.

A alternate way to make the parachute top is to create a UV sphere, and cut it in half.

61. Model a Low Poly Head

UNKNOWN TEMPLATE blender3d_Aligned_to_view_issue

Warning

Using triangles on a subsurfed model may result in "peaks" appearing in some areas. To reduce this problem, merge as many triangles into quads as possible.

61.1. Overview

This tutorial is designed to teach users to make a low-poly anime head in Blender.

What you need to know:

- Basic Blender controls

61.2. Add a plane

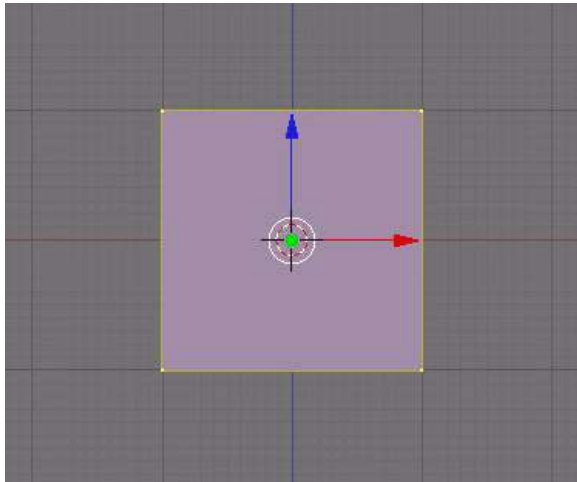


Figure 452 Add a plane

Start with a new file, and delete (**X**Key) the default cube.

Be in **Front** view (**NUM1**)

Add a plane (spacebar-> add-> mesh-> plane)

61.3. Make a pointed chin



Figure 453 Merge the vertices

Switch to **Edit mode**

Be in **Vertex select** mode

Select the bottom two vertices (**SHIFT+RMB**) and press **W** to bring up the vertex menu. "Select Merge", then "At Center", or just hit **ALT+MKEY** and choose "At Center".

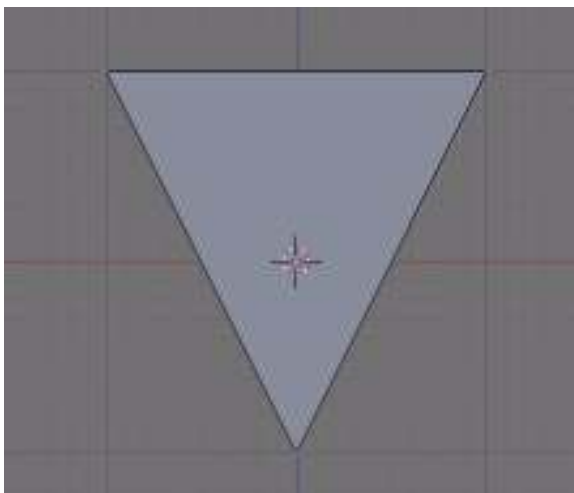


Figure 454 A pointed chin!

Now you have a pointed chin.

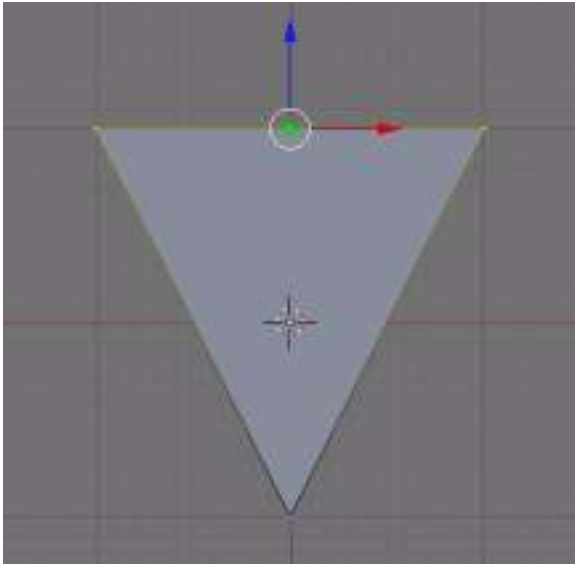


Figure 455 A pointed chin!

Select the top two vertices (use the **BKEY**)

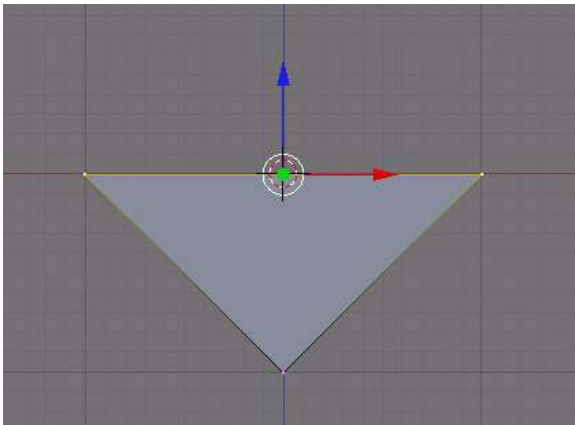


Figure 456 Select the top 2 verticies

Re-arrange them so they make more of a chin shape by pressing the **GKEY** to move and **ZKEY** to constrain the movement to the Z-Axis.

Noob Note: Move it down; don't scale it out.

61.4. Extrude the face

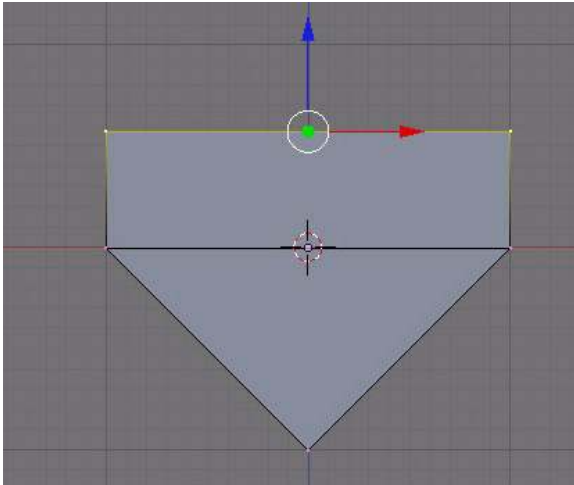


Figure 457 Extrude along the Z-Axis

Now extrude (**EKEY**) "edges only", along the Z-Axis (**ZKEY**) so that you have another area.

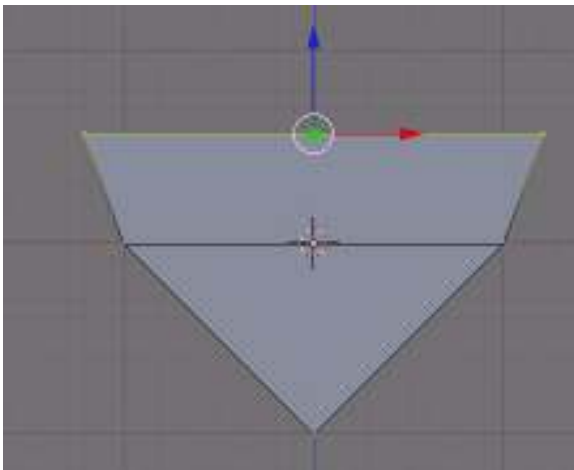


Figure 458 Scale it

As a precaution, remove doubles. (**WKEY**)

Scale it (**SKEY**) so that it's not so cubic.

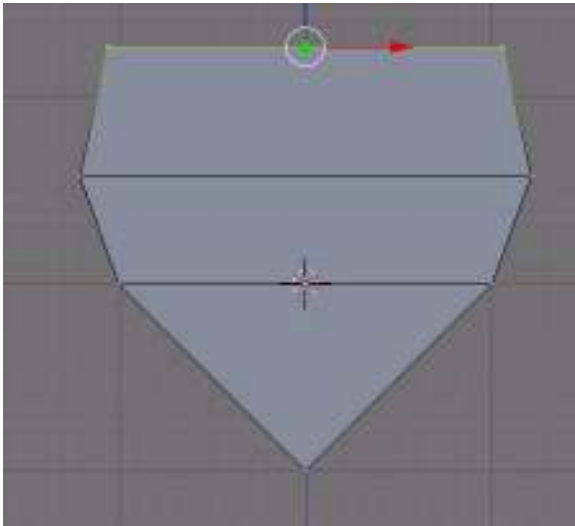


Figure 459 Extrude along the Z-Axis again

Now extrude along the Z-Axis again (**EKEY** then **ZKEY**) and scale (**SKEY**) it down a bit.

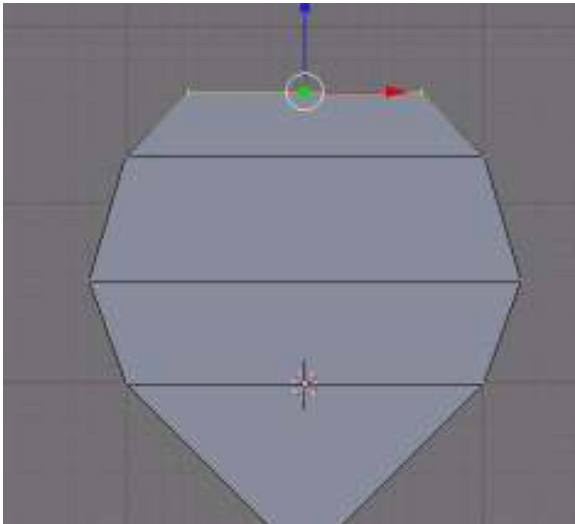


Figure 460 Extrude along the Z-Axis again

One last time... (E) (Z) (S)

61.5. Make the facial features

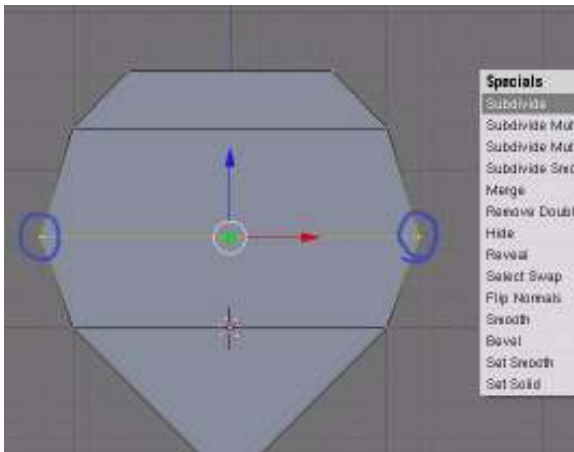


Figure 461 Select the middle vertices

Select the middle vertices (**BKEY**) or (**RMB**) and press (**WKEY**) to bring up the specials menu. Subdivide it once.

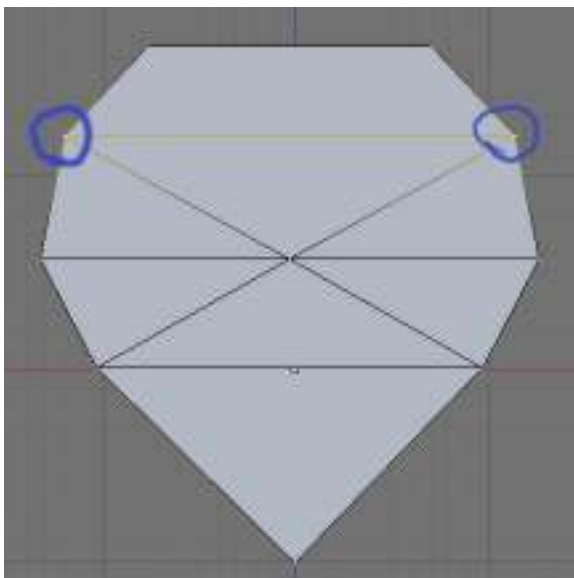


Figure 462 Select the vertices

Now select at first the pair of vertices above and subdivide once.

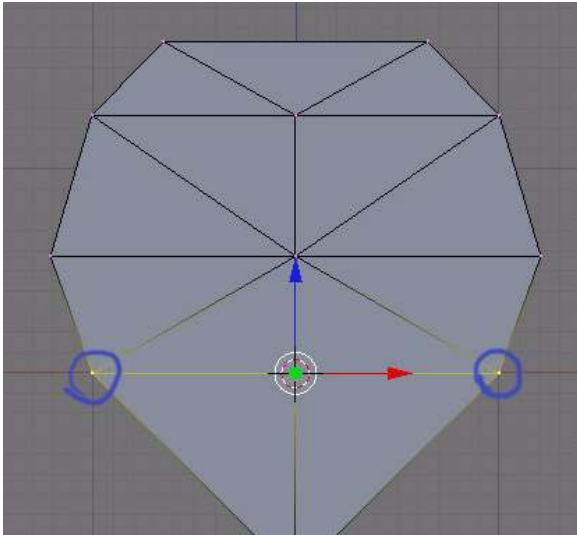


Figure 463 Subdivide!

Now subdivide 3 times with the pair of vertices below the center line.

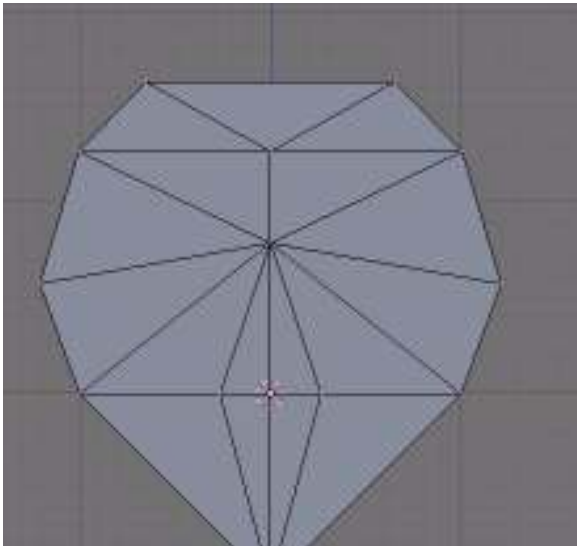


Figure 464 There!

Yours should now look like this one on the right.

61.5.1. Eyes

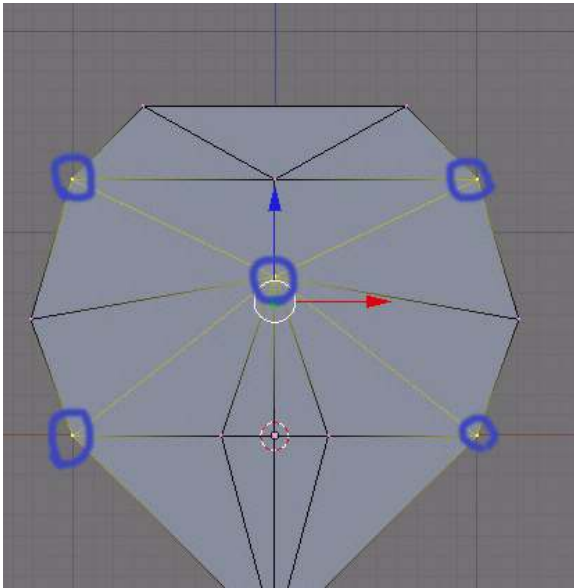


Figure 465 Subdivide some more

Select and subdivide here, so you can make the eyes.

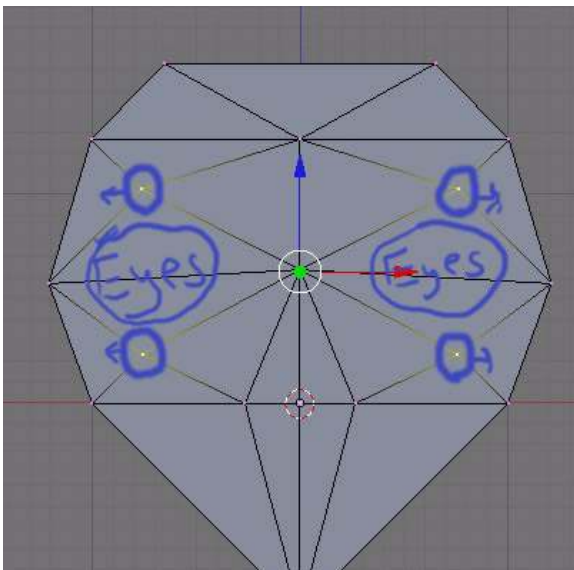


Figure 466 Make some eyes

And rearrange the vertices so they make eye shapes.

61.5.2. Nose

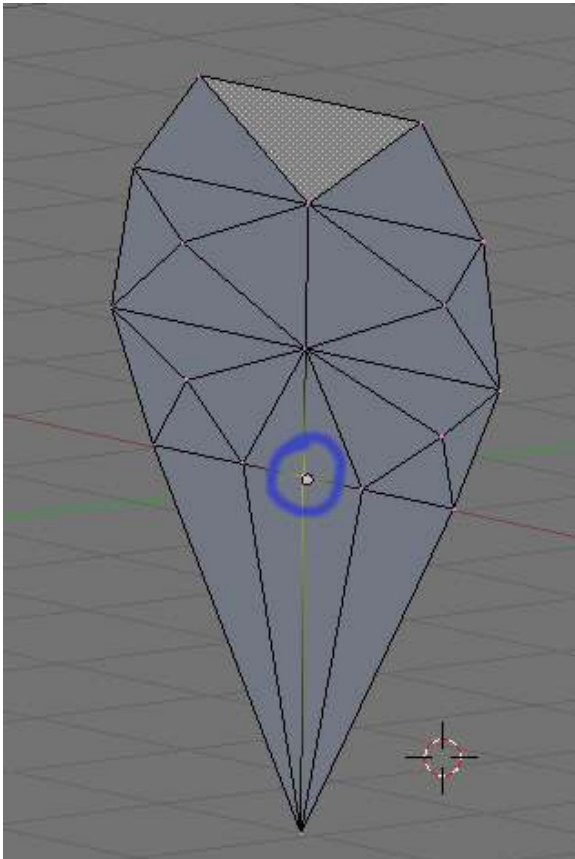


Figure 467 Select these

Now select the middle vertex here (right mouse button).

Extrude it along the Y-Axis (**EKEY**) (**YKEY**) and move your mouse around to change how far it moves.

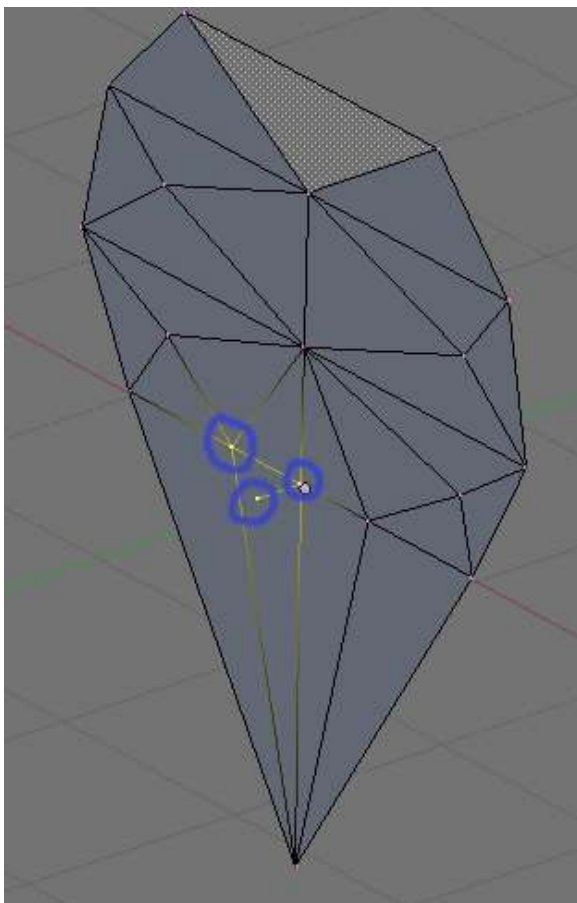


Figure 468 Let's make a nose!

Select these three vertices (**SHIFT+RMB**) *and press (F) to* make a face.

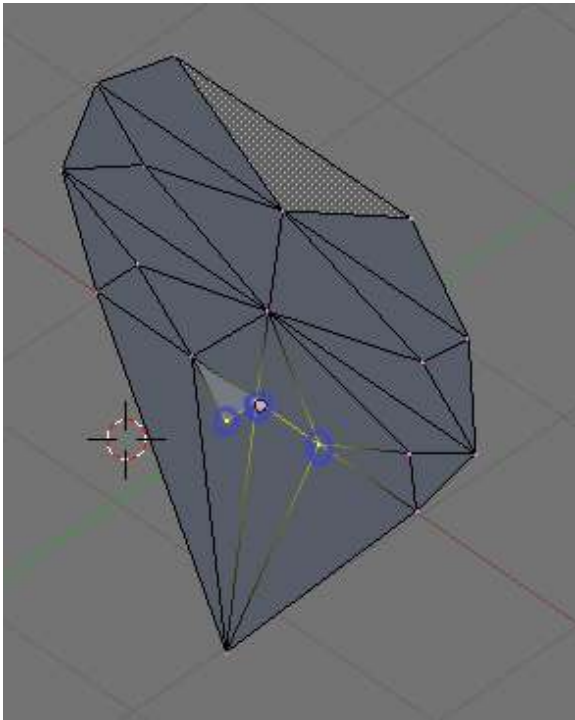


Figure 469 Do the same to the other side

Now do the same to the other side, and you will have the nose's base.

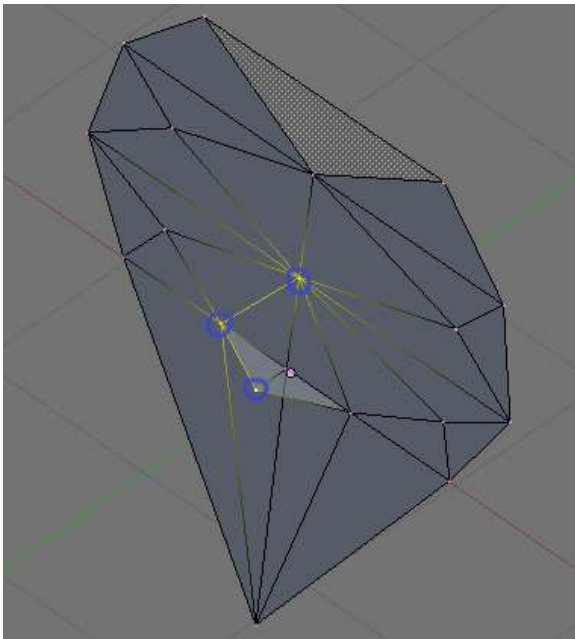


Figure 470 Make a face

Select these vertices (**SHIFT+RMB**) and make a face (**F**).

Do this to the other side.

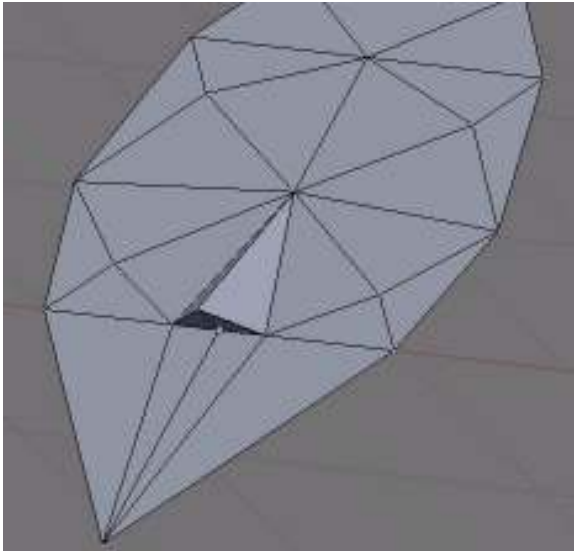


Figure 471 You have a nose!

Now you have a nose.

61.5.3. Remove interior nose faces

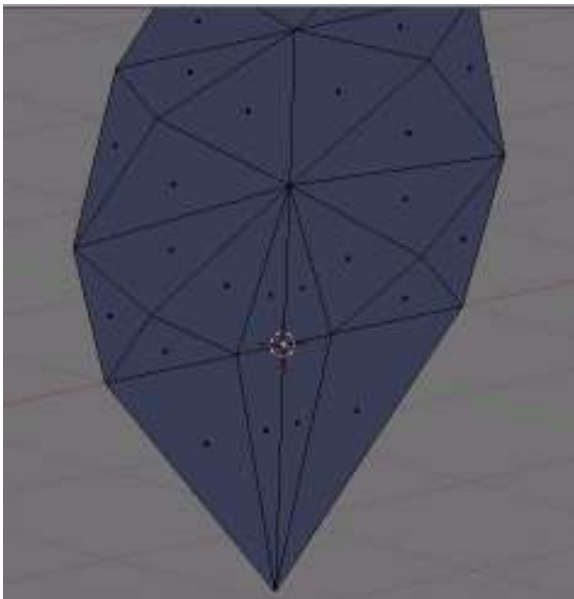


Figure 472 Rotate the camera

Rotate the camera (**CTRL+NUM1**) so you are seeing the back.

Be in **Face select** mode.

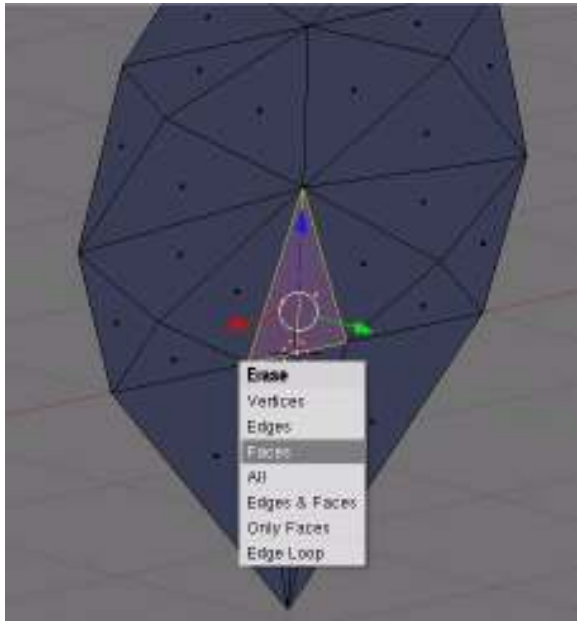


Figure 473 Get rid of these

Select (**SHIFT+RMB**) the faces behind the nose.
Delete (**XKEY**) them (on the popup pick "Faces").

61.6. Smooth the mouth region

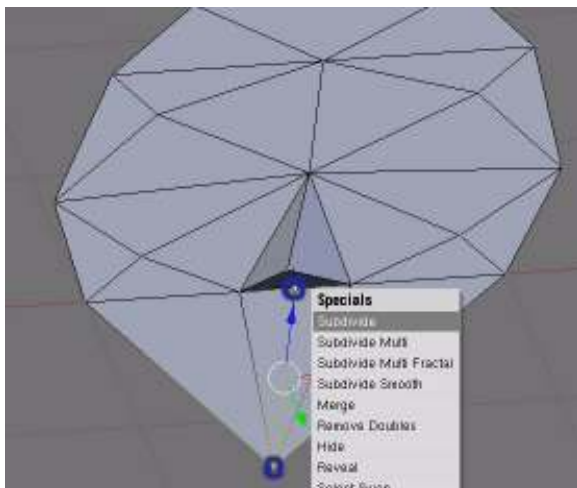


Figure 474 Select these

Be in **Vertex select** mode

Select the vertex at the bottom of the chin, and the one at the bottom of the nose.

Subdivide them.

61.7. Make the head

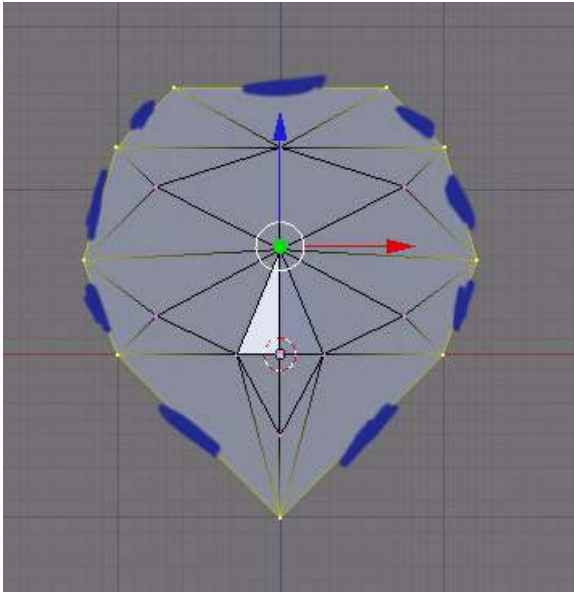


Figure 475 Select edges on the outside

Be in **Edge select** mode

Select all the edges on the outside (shown with blue lines in the photo).

UNKNOWN TEMPLATE Clr

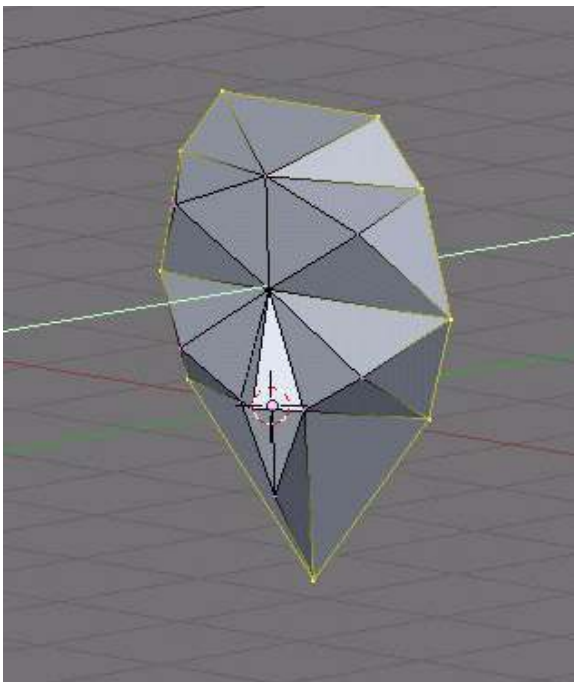


Figure 476 Give the face more smoothness

Move (**GKEY**) the vertices back, along the Y axis (**YKEY**), a bit to give the face more smoothness

UNKNOWN TEMPLATE Clr

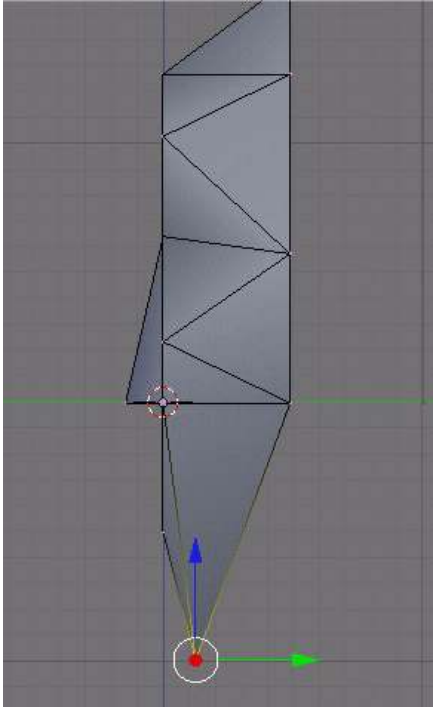


Figure 477 Select the vertices at the back of the head

Be in **Vertex select** mode

Rearrange the chin, by moving (**GKEY**) the bottom vertice along the Y axis (**YKEY**) till it looks right.

61.8. Make the back of the head

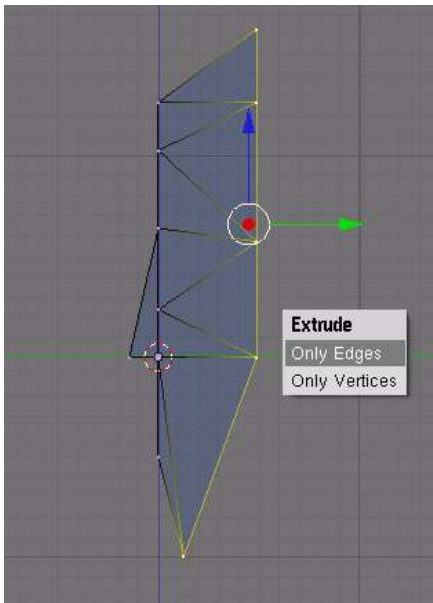


Figure 478 Extrude along Y-Axis

Be in **Edge select** mode

Now select the edges all around the back of the head, using (**BKEY**).

Extrude these along the y axis (**EKEY**) (**YKEY**)

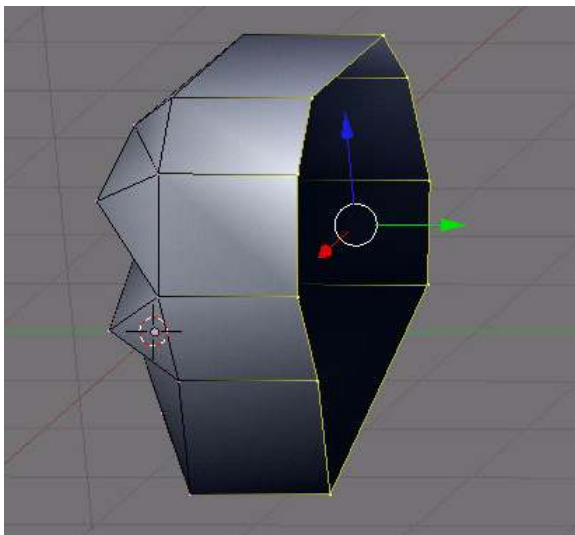


Figure 479 Ta da!

Ta da!

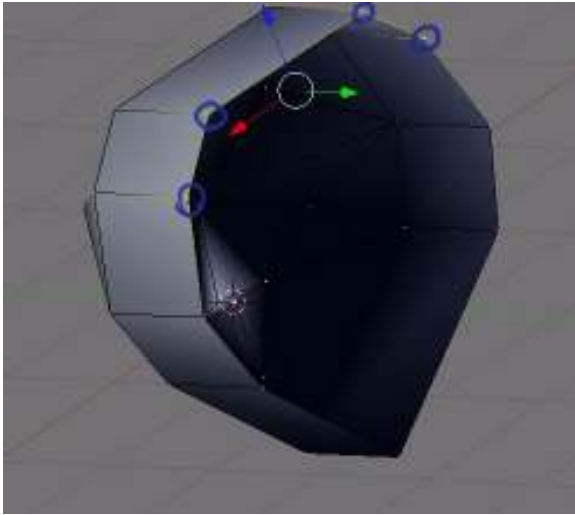


Figure 480 Make a face

Now select a set of 4 back vertices that form sort of a rectangle and press **(FKEY)** to make a face.

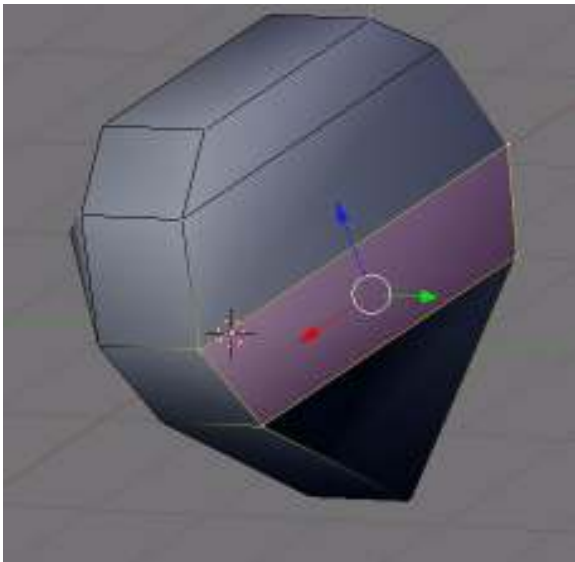


Figure 481 and so on...

and so on....

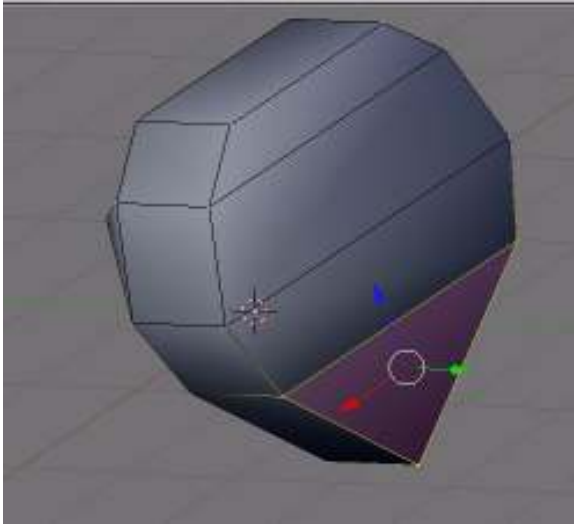


Figure 482 and so on...

and so on....

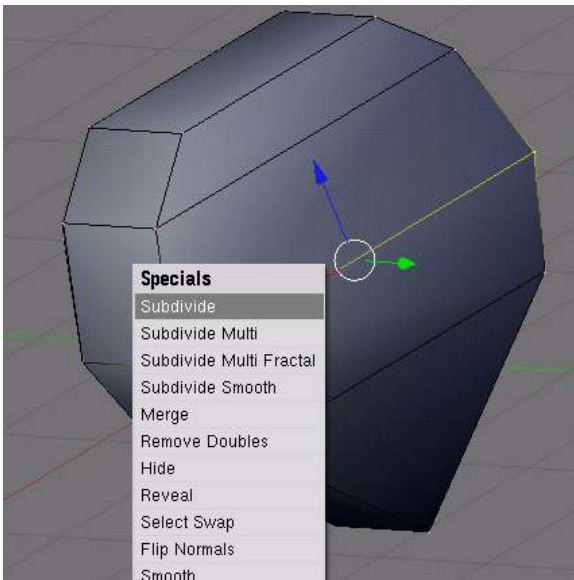


Figure 483 Subdivide these

now select and subdivide these once (W)

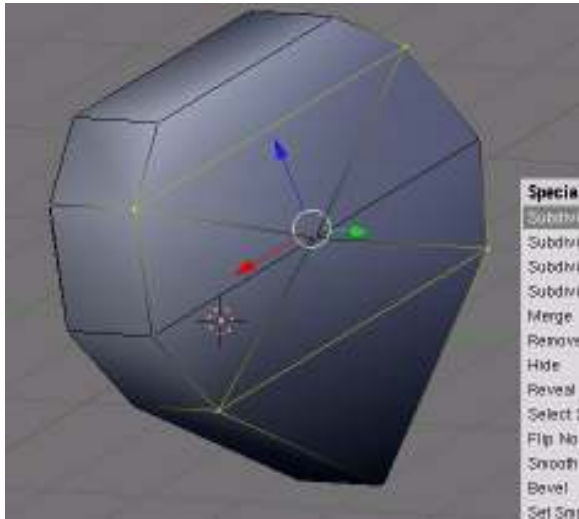


Figure 484 Subdivide these as well

and these points too.

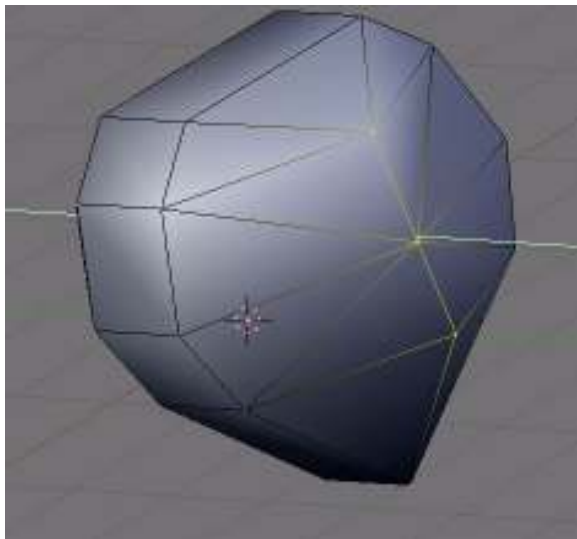


Figure 485 Move the middle back

Now select the middle and move it back (G) (Y).

61.9. Finishing it up

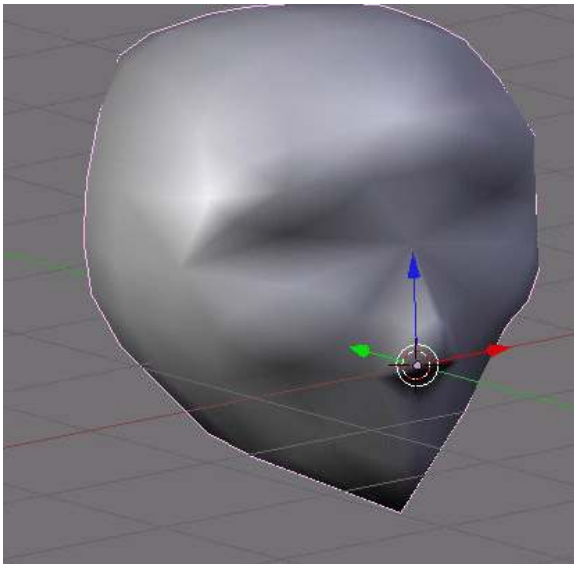


Figure 486 With subsurfing

And here it is with sub-surfacing. You have finished. Hit (F12) to see the final render.

62. Building a House

This tutorial is based on Bart Veldhuizen's "Building a House" from Tutorial #01 published by NaN in 1999 which is also available in a PDF at <http://download.blender.org/documentation/BlenderTutorialGuide1.tar.gz>¹ (a tar.gz containing BlenderTutorialGuide1.pdf) or <http://download.blender.org/documentation/BlenderTutorialGuide1.zip>² (a ZIP file containing the same PDF).

Permission was asked to use it and Ton Roosendaal said "Be assured that everything that was produced by NaN now is open and free content for everyone to reuse, including the tutorial "Building a House"."

i Information

Applicable Blender version: 2.49b

62.1. Setting the Scene

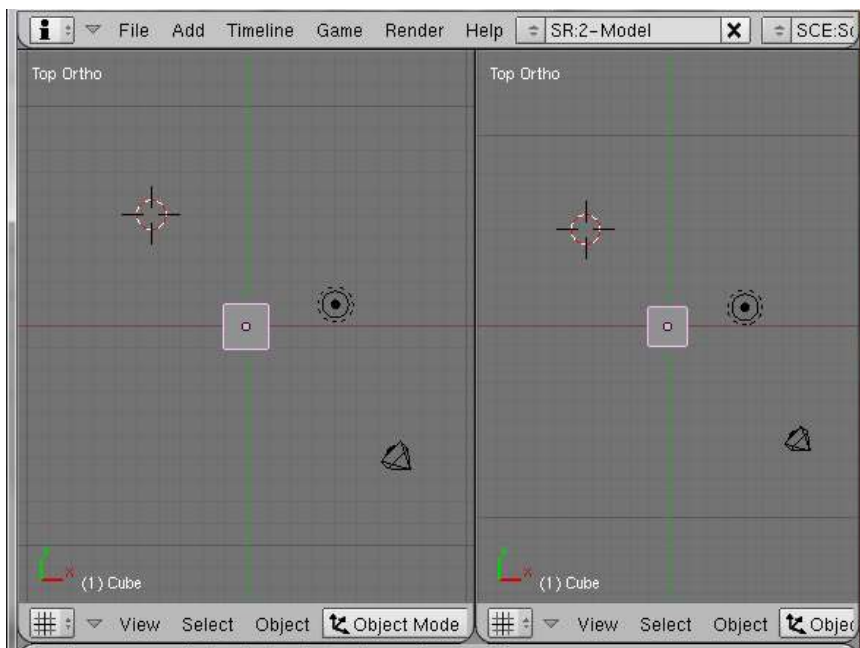


Figure 487

1 <http://download.blender.org/documentation/BlenderTutorialGuide1.tar.gz>

2 <http://download.blender.org/documentation/BlenderTutorialGuide1.zip>

Start a new document (**CTRL** + **X**)

Split the window into two side-by-side windows: move the mouse to the line between the 3D window header and the Button window where it becomes an up-down arrow.

Press **RMB** and select "Split Areas" from the menu. Move the line which appears to the middle and click **LMB**.

UNKNOWN TEMPLATE Ctr

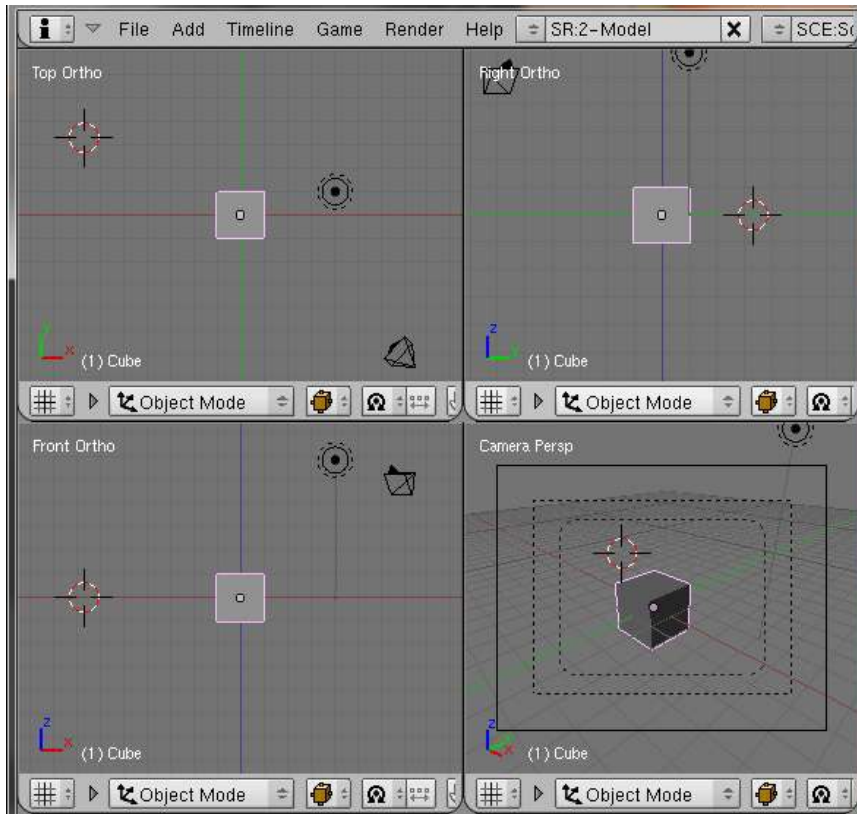


Figure 488

Now move the mouse to the line separating these two windows,

click **RMB** again and split the left-hand window into top and bottom windows.

Do the same to the righthand window to get four windows.

Make the upper left hand window into Top view (**NUM7**);

Make the lower left hand window into Front view (**NUM1**)

Make the upper right hand window into Side view (**NUM3**);

Make the bottom right hand window into camera view (**NUM0**);

UNKNOWN TEMPLATE Ctr

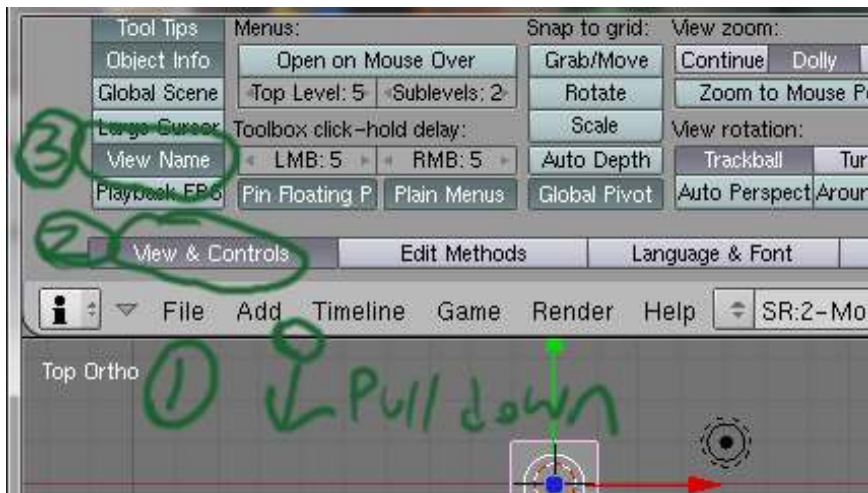


Figure 489

Now before you forget which direction each window is looking at, put the cursor over the lower edge of the top menu bar, and hold **[LMB]** as you pull it down, to expose the setting panel.

Find the **View name** option, turn it on, and now each of the four views will have the direction it's looking at, listed in the top left, of each window.

While you're here you should take care of the align to view setting as shown in the warning box at the top of this page.

62.2. Make the house walls

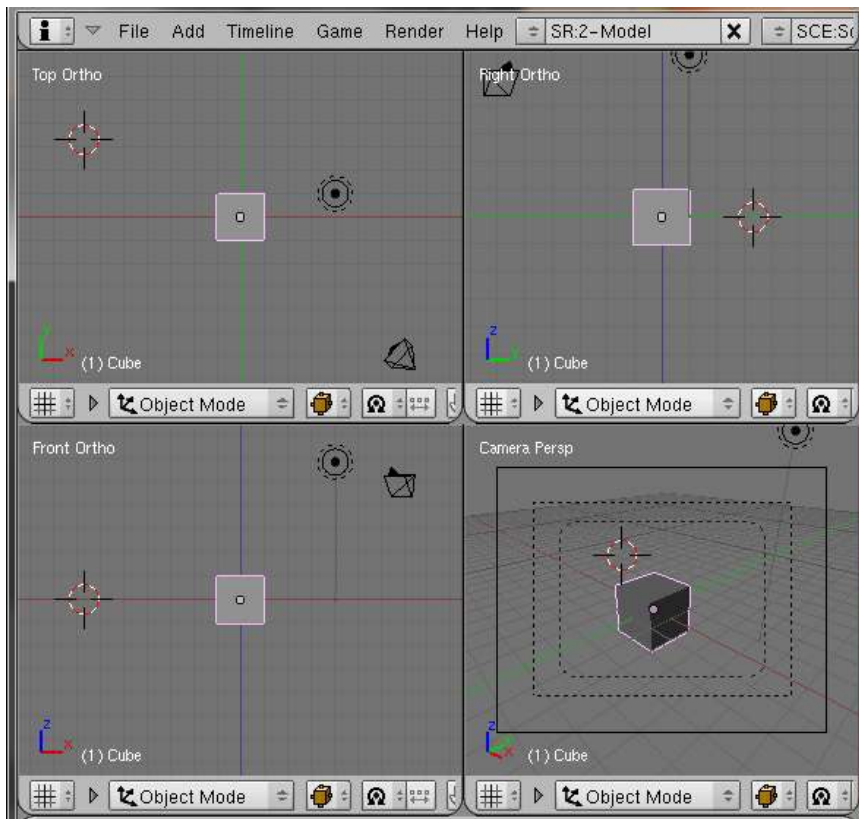


Figure 490

Leave the default cube, it will be the walls of the house.

62.3. Make the roof

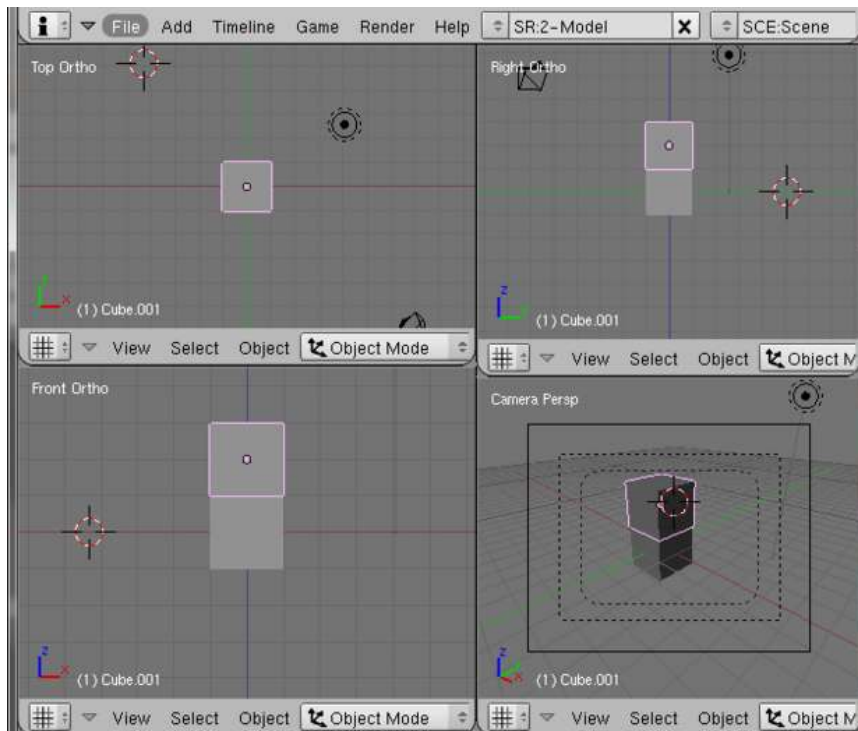


Figure 491

Ensure you are in **object** mode (**[TAB]** switches between object and edit mode)

Now, in the Front view window (lower left)

Press **[SHIFT] + [D]** to duplicate the cube, grab mode is automatically selected, press **[Z]** to restrict the move to the up-down axis and move the cube to rest on top of the original (pressing **[CTRL]** while moving will snap it to the grid and make it easier to position accurately.) Press **[ENTER]** when it is in place.

You can see how useful this four-paned window is; it shows you exactly what's going on.

Noob note-- Make sure that you are in object mode when you duplicate the cube to create your roof. It needs to be its own object so that you can edit it separately from the house object, since you will put one through the other later on (to get the eaves to extend past the walls of the house, without doing extra extrusions)

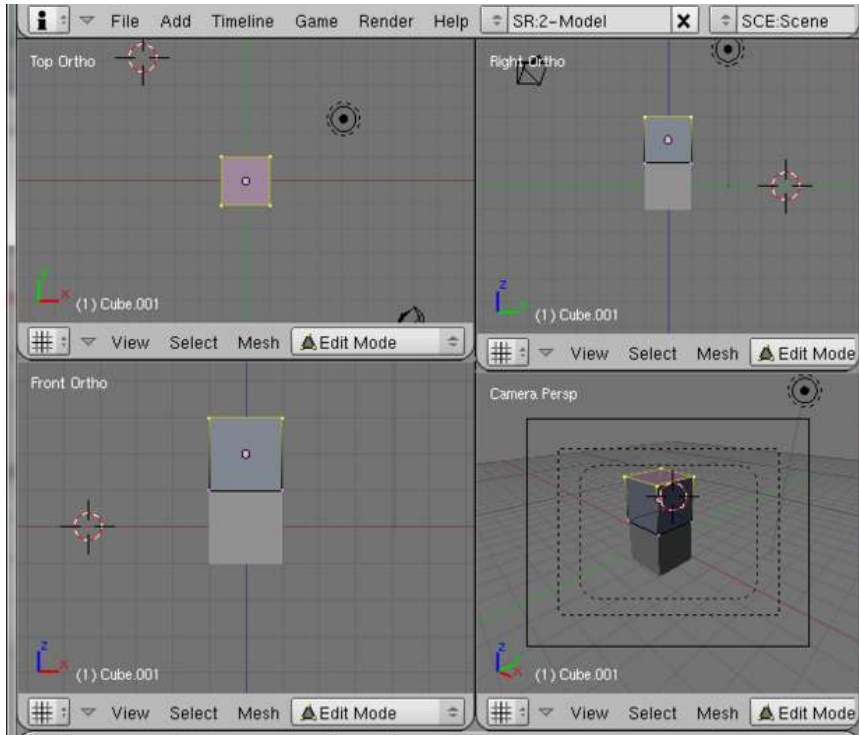


Figure 492

The top cube is going to become the roof and needs to be made into a triangle

Select the top cube (if it is not already selected) by clicking **[RMB]** on it

Press **[TAB]** to go into **Edit Mode** (check the box in the middle of the 3D window header)

Press **[A]** to deselect all the vertices

In the front view, or side view, box select (**[B]**) the top four, you will be able to see that they have been selected in the camera view (lower righthand) window.

UNKNOWN TEMPLATE Clr

62.3.1. Make the apex of roof

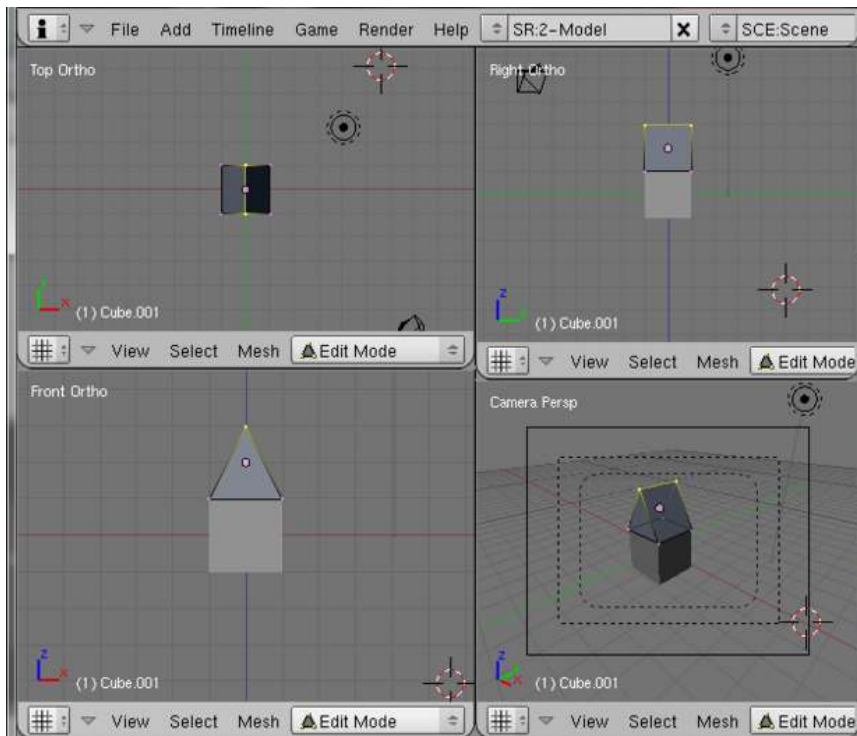


Figure 493

Now press **[S]** for scaling and **[X]** to limit the scaling to the X-axis (left to right). You could try to move the mouse back and forth to bring the top vertices together, but you will have a hard time lining them up exactly, so it's best to just type in 0 followed by **[ENTER]** to set scaling to zero.

Note: Another way, although more steps, to choose 2 vertices on one side, and merge them at Center (**[ALT] + [M]**, choose "Center" and click the Window). Repeat this with the other 2 vertices and you'll have the same result.

62.3.2. Scale the roof

The roof also needs to project over the walls of the house, this can be done in two ways: either scale the whole roof making it taller as well as wider and longer front-to-back or it can be left the same height but just scaled along the X and Y axis.

Noob note: the second method is better

Scaling method 1

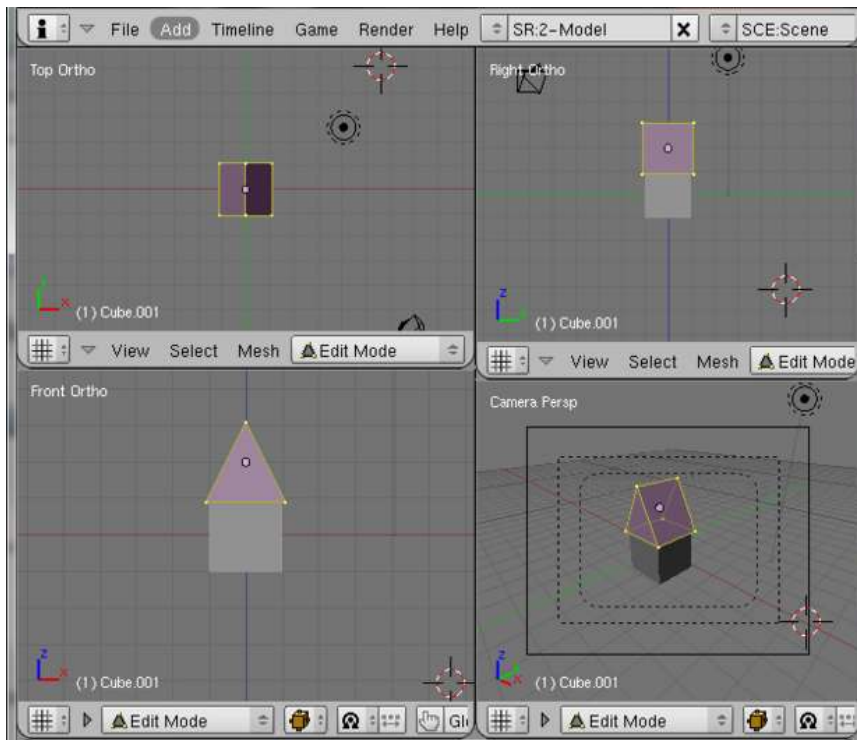


Figure 494

This way is simpler, but changes the proportions of the roof, which you may not want to do in a project

Zoom in (**MMB**) if you need to.

Press **A** to deselect everything

Then place the mouse over the roof and hit **L** to select the whole roof.

Scale **S** the whole roof to about 1.1.

Scaling method 2

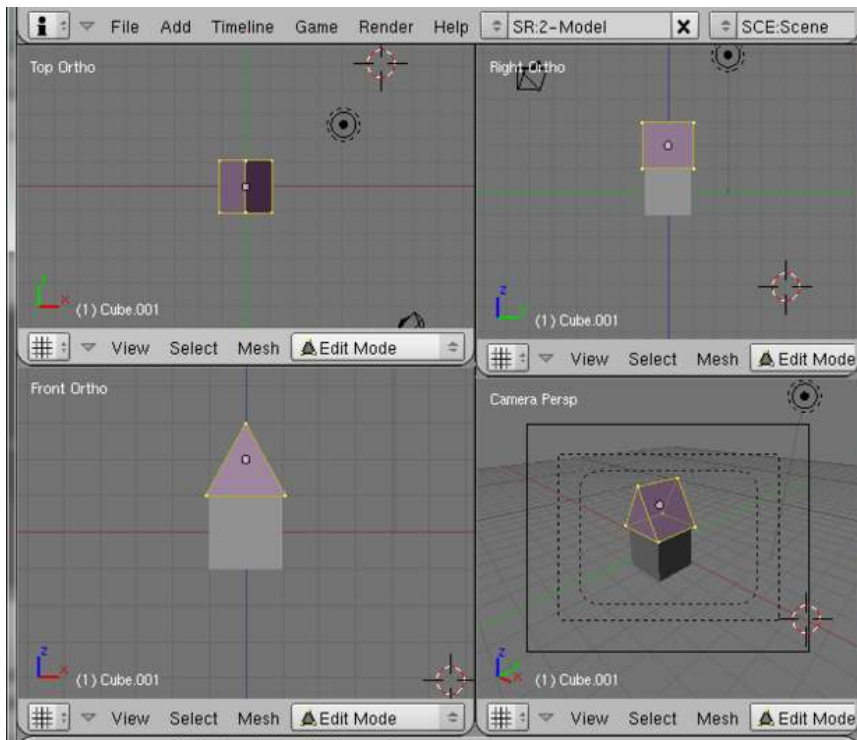


Figure 495

To change the roof leaving the height the same:

In the Front view (lower left) window press **A** to deselect everything.

Then place the mouse over the roof and hit **L** to select the whole roof.

Press **S** and **SHIFT** + **Z** and scale to 1.1

62.3.3. Naming the roof and house objects

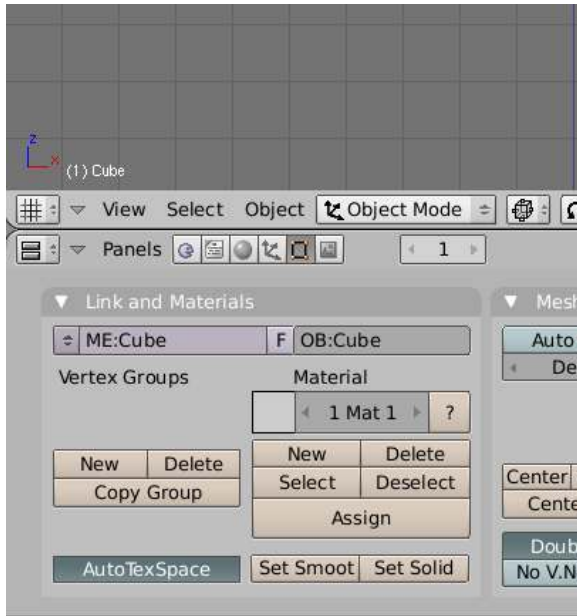


Figure 496

It is helpful to change the names from the default names to ones which describe the objects: for instance, you could change the name "Cube" to "House" instead and "Cube.001" to "Roof".

However, there is an object called "Cube" and a mesh called Cube which is linked to it so there are two names to change. There is also a material linked as well.

To change the name of the house object, go into object mode (**TAB** if not already there), select the original cube, select the Editing panel from the Button window header (**F9** or click on the icon) and, in the panel "Link and Materials", you will see

"ME:Cube" and "OB:Cube". Click on "Cube" and it will change to a red background, type in your new name, for instance "House" and press **ENTER** . Do this for both.

Now select the roof cube in the 3D window and go through the same procedure renaming it from "Cube.001" to "Roof".

62.3.4. Coloring the house

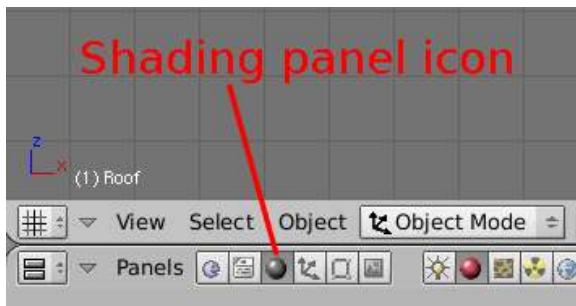


Figure 497

The roof now needs to be given a different colour to the default grey. Select the Shading panel from the buttons window (**F5**) or click on the icon)

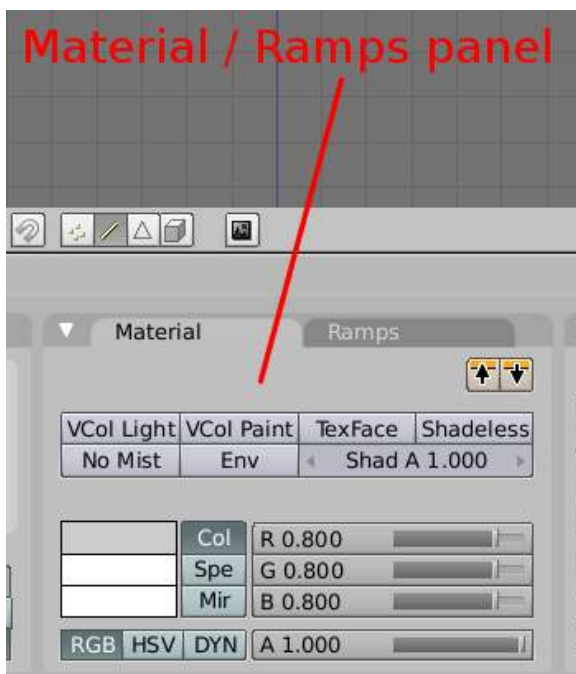


Figure 498

In the "Links and Pipeline" panel, click on "MA:Material" in the box under the "Link to Object" title. "Material" will appear on a red background, rename it by typing "RoofRed" and pressing **ENTER**. You can now go to the second panel with two tabs "Material" and "Ramps" ("Material" should be showing, otherwise click on the Material tab to get it to show).

Note: In v2.45 you will not see any MA:Material box because the program does not assign any material to a new object. You will first have to click the Add New button to display the material info.

Now go to the second group of buttons and the line with "Col" in the second column and click on the grey bar to its left. This will bring up a colour selector, select a suitable red and press **ENTER**:

lo and behold, the whole house turns red. If nothing happens you are probably in wireframe mode: press **Z** in all four windows to change to solid mode.

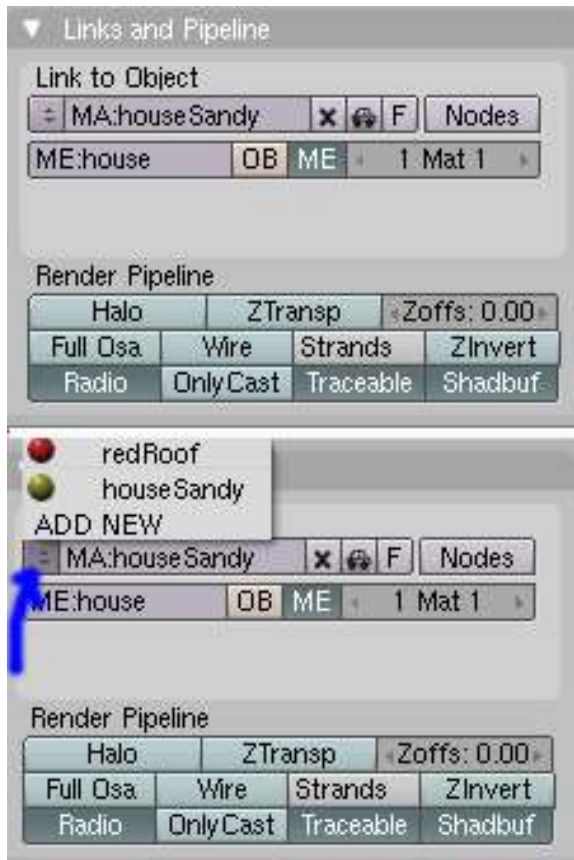


Figure 499

To change the house colour to something more appropriate, go back to the lower left window and select the house cube, in the "Links and Pipeline" panel the object will change to "House".

At present, both cubes share the same material, so, to colour the house cube differently, a new one has to be created.

Click on the up-down arrow to the left of "MA:RoofRed" in the "Links and Pipeline" and select "ADD NEW", the text to the right of this arrow now changes to "MA:RoofRed.001".

Click on "RoofRed.001" as before and change it to "HouseSandy". Now click on the bar coloured red next to the word "Col" in the "Materials" panel and you will get a colour selector, choose a suitable colour for the house, press **ENTER** and the house cube will now take the colour chosen.

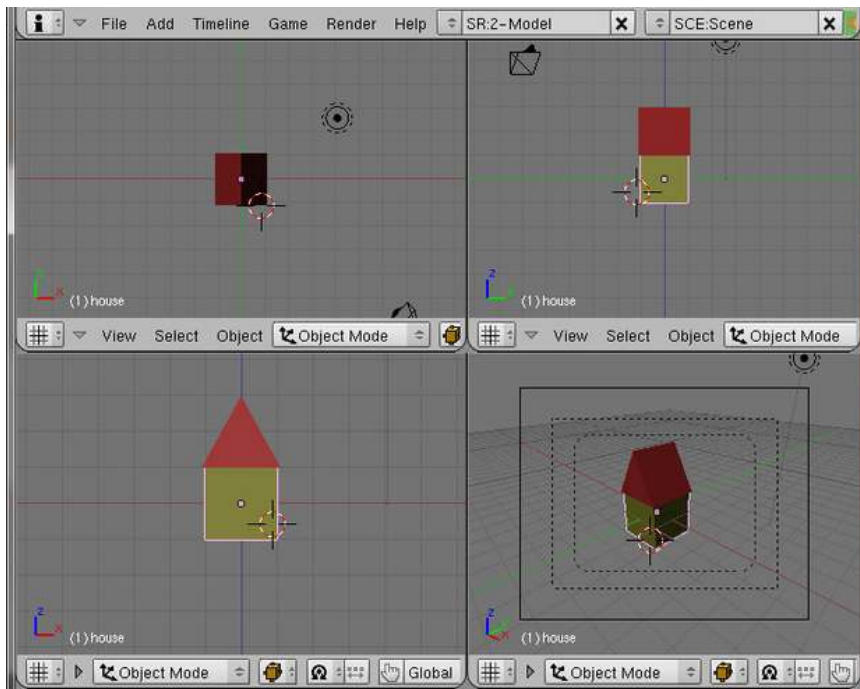


Figure 500

your house should look like this now

62.4. Make a chimney

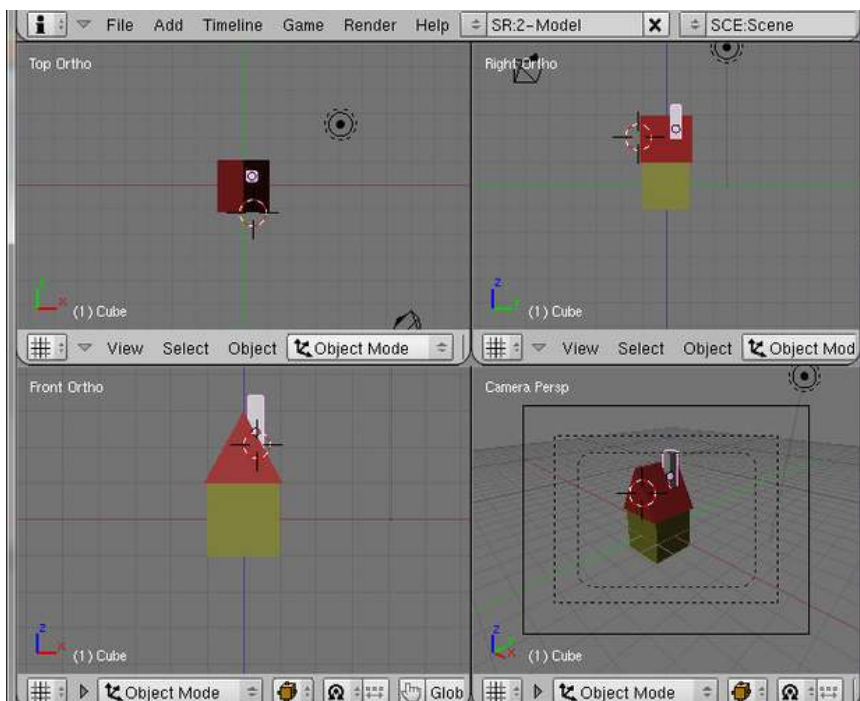


Figure 501

Go into Object mode, if you're not already in it. (**TAB** flips between object and edit mode.)

Be in Front view (lower left window).

Click **LMB** on the righthand side of the roof, this will select the spot where the chimney will be created.

Press **SPACE** and select Add -> mesh -> Cube from the popup menu — a bit large, isn't it?

Scale it along the X and Y (**S** , **SHIFT** + **Z**) to about 0.2.

Now grab it (**G** , **X**) and move it into position looking at the front view (lower left window).

When this is correct, move it into position along the Y axis (**G** , **Y**) looking at the top view (upper left window).

You can check the side view (upper right window) and the camera view (lower right window) to see that it looks OK.

UNKNOWN TEMPLATE Clr

62.4.1. Name and color the chimney



Figure 502

In the link and materials panel in the Edit button panel, change the name of the chimney object and the mesh linked to it to "Chimney".

Next in the "Links and Pipeline" panel in the shading button panel, you will notice there are no buttons, and there is a drop down box with the words "Add New" in it. You can now choose to have the chimney a different colour to the roof or the house or reuse one of these colours.

If you want a new colour, click **LMB** on the "Add New" button and select it as you did for the house and the roof. Rename the material.

If you want to reuse the same colour, you can click **LMB** on the up-down arrow to the left of MA:Material and select from the choices "RoofRed" and "HouseSandy".

62.5. Adding a Window and a Door

62.5.1. Make a window frame

Make the frame

Be in Object mode.

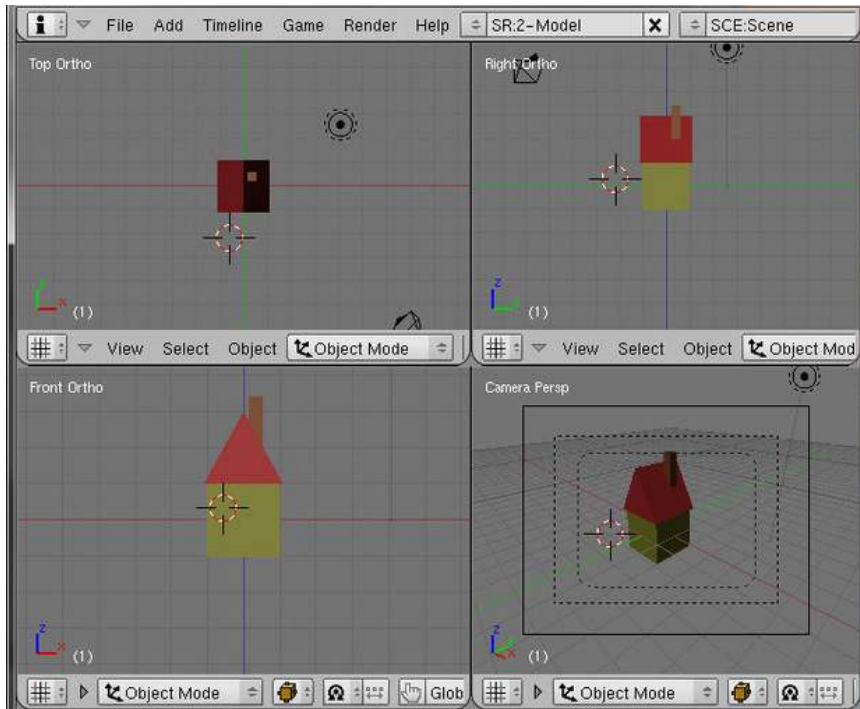


Figure 503

Using the front and top views, click **LMB** to place the 3D cursor where you want the window to go.

In the front view (lower left) add a plane (**SPACE** Add > Plane).

Noob note: if the plane shows up laying flat, then rotate it into the right orientation: **R** **X**, enter 90 for the angle, then **ENTER**.

Scale **S** it down to 0.4, and press **ENTER**

To give it a rectangular shape, scale the plane by about 0.8 along the X-axis **X**.

Move it into position (**G**), using the top and front view, on the left hand side of the house a little higher than midway.

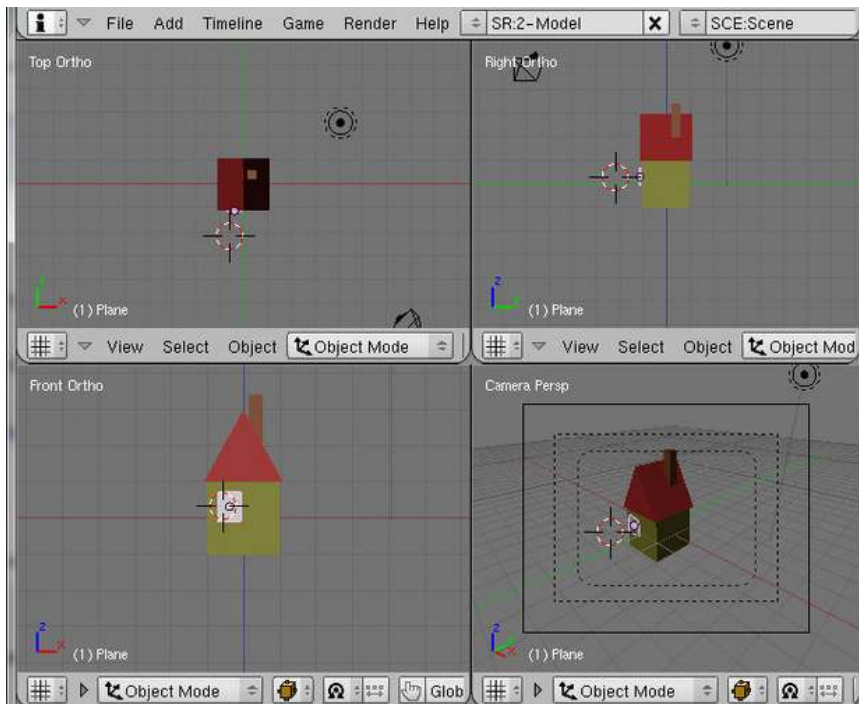


Figure 504

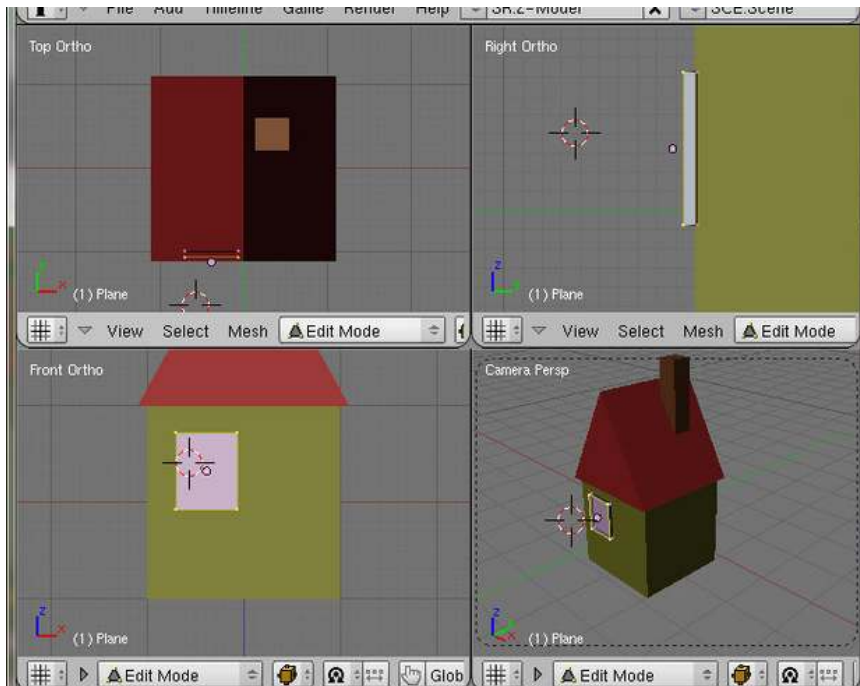
Be in the Side View window (upper right).

Move the plane along the Y-axis (**G** , **Y**) until it is just in front of the house. You may want to zoom in (**MW**) to see better.

Now, still in Side View,

Switch to **Edit** mode (**TAB**)

Extrude the plane towards the house (**E** , "Region", -0.07) until it is embedded in the wall.

Make the sill**Figure 505**

To make this box into a window frame will require a larger view: move the mouse into the Side View window and zoom in by rotating the **[MW]**. Optionally, you can press **[NUM]** to see just the window object, and then **[CTRL] + [UPARROW]**, to make the view your mouse is over become full screen.

Deselect all (**[A]**) and select the outer (leftmost in the right view) vertices using box select (**[B]**), press **[E]**, select "Region" and immediately (without moving the mouse) press **[ESC]**.

Scale the newly created vertices by 0.9 (**[S]**, 0.9, **[ENTER]**).

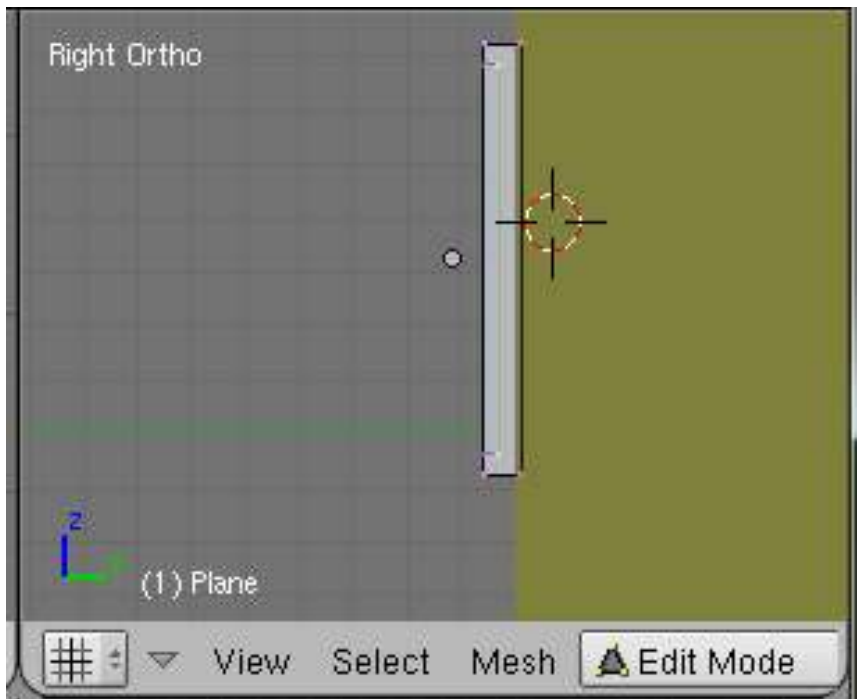


Figure 506

Extrude again inwards to most of the depth of the box about $-.03$ to $-.05$ along the Y axis, depending on how you have your window into the wall(**E** , "Region", **Y** , $-.03$, **ENTER**).

Noob note: the outer panel of the window that you just made cannot be inside the wall, you will see this best on the side view. (if the outer face of the window pane is inside the wall you either won't see any paint/texture you apply, or it will half show up.)

If you used the object view, or single window views, then press **CTRL** + **UPARROW** and **NUM/** to get back to your original four views.

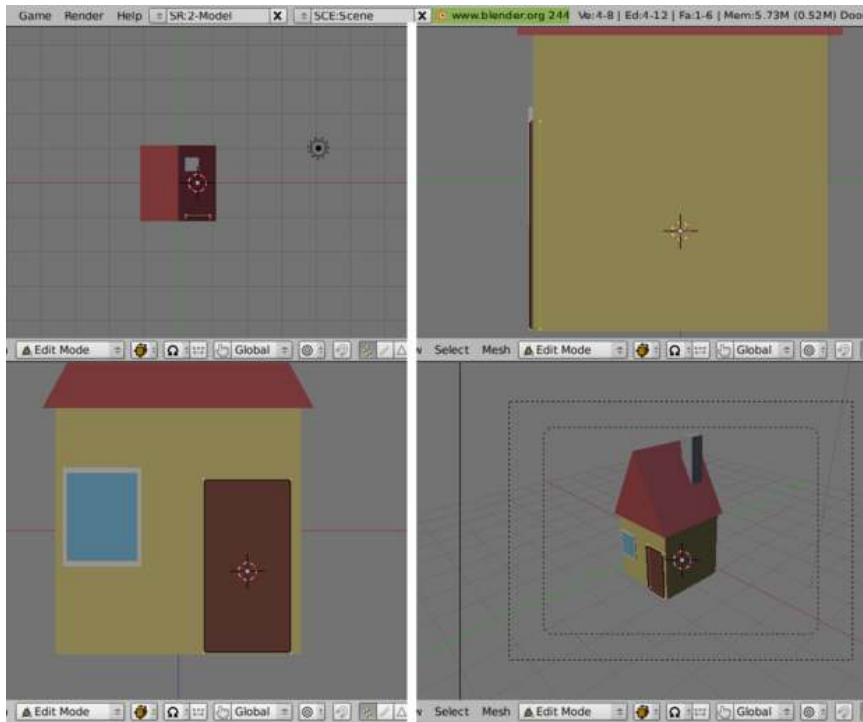
Naming and coloring the window frame object

To change the name of the object and the mesh to "WindowFrame": go into Object mode, select the Editing panel (**F9**) and, in "Links and Materials" change "ME:Plane" and "OB:Plane" to "ME:WindowFrame" and "OB:WindowFrame". Under "0 Mat 0" click on the "New" button to assign a material to the window frame: when this changes to "1 Mat 1", click on the grey bar to the left of it and select white from the Colour Selector which appears.

Select the shading panel (**F5**) and change the name of the material ("MA:Material") to "WindowFrame".

Make window glass

Method 2 is an easier way to make the window glass

Method 1 (Create a Separate Window Pane Object)**Figure 507**

Deselect all (**A**),

Create a plane. Ensure it is vertical (**R** **X** 90 **ENTER**).

Scale it to the dimensions of the window frame.

Move it in front of the frame.

Be in Side view.

Move it to the front of the window frame.

Be in front view.

Be in full screen (**CTRL** + **UPARROW**)

Adjust the position and size so it fits within the frame.

Naming and coloring the window glass object

Go back into Object mode (on older versions creating a new object automatically puts you into Edit mode)

Change the mesh and object names to WindowGlass.

Add a new material for the glass and colour it light blue (I chose "87ceeb" SkyBlue).

Method 2 (Assign a Glass Material to the Pane Part of the Window Object)

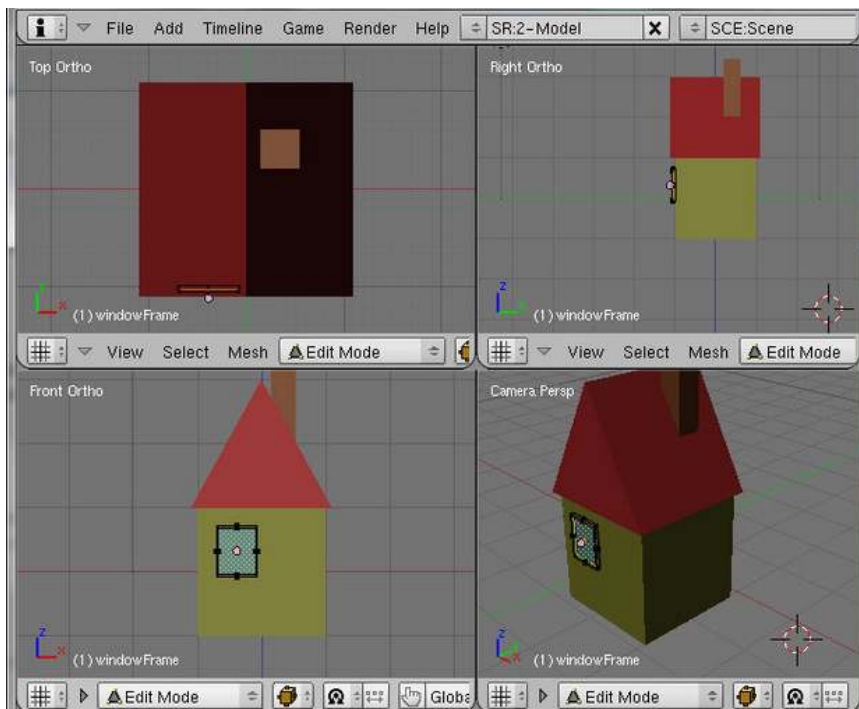


Figure 508

Deselect all (**A**)

Be in **Edit** mode

Switch into **Face** (**CTRL** + **TAB** -> faces) select mode (you've been in **Vertex** select mode up to now)

RMB on the face in the center of the window frame, to select it.

Hit the "New" button, under the 1 mat 1 material box in the Link and Materials panel, you should now have 2 mat 2 displayed in there.

Now to connect the new material to just that one face, press the assign button under the new button.

Naming and coloring the window glass face

You can now change the color by clicking in the square to the left of the "2 mat 2".

Now go to the links and pipeline panel on the shading panel (**F5**) and rename the new material to windowPaneGreen.

Now switch back to **vertex** select mode (**CTRL** + **TAB** -> "vertex").

62.5.2. Make the door

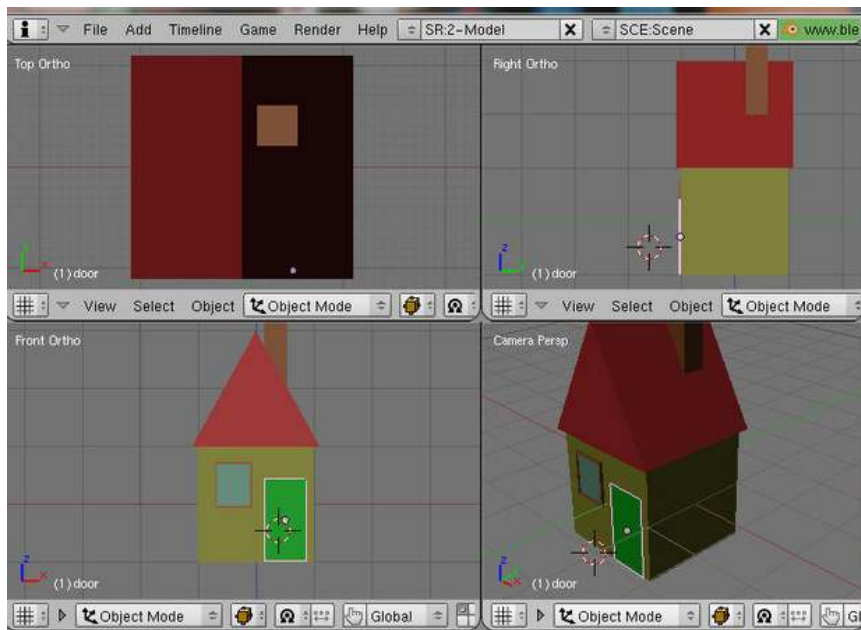


Figure 509

Add the door using the same method:

Be in Object mode

Deselect all,

Create a plane, rotating it as necessary (**[R]** **[X]** 90 **[ENTER]**) to align it with the front wall of the house.

Scale the plane down by .7 and then again by .5 along the X axis.

Move it into position using Front view and then move it against the wall of the house in Side view.

Naming and coloring the door object



Figure 510

Change the Object name and the mesh name to "Door"

Create a new material for it

Change the material's name to "Door" and chose an appropriate colour for it.

UNKNOWN TEMPLATE Clr

62.6. Building the Fence

The way to build the fence is to create a paling and duplicate this as often as necessary.

62.6.1. Make a paling

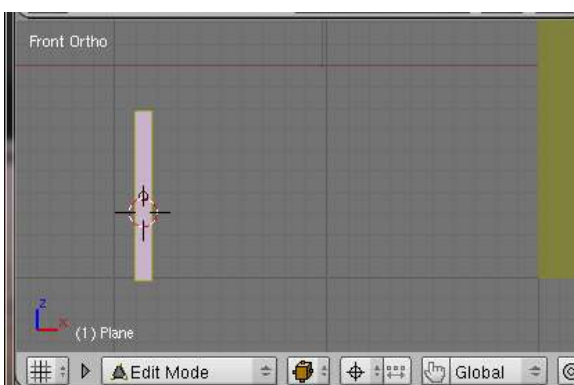


Figure 511

Be in object mode, with nothing else selected.

Add a plane, orient it vertically (**R** **X** 90 **ENTER**), scaled to 0.4 and then 0.1 along the X axis in Front view.

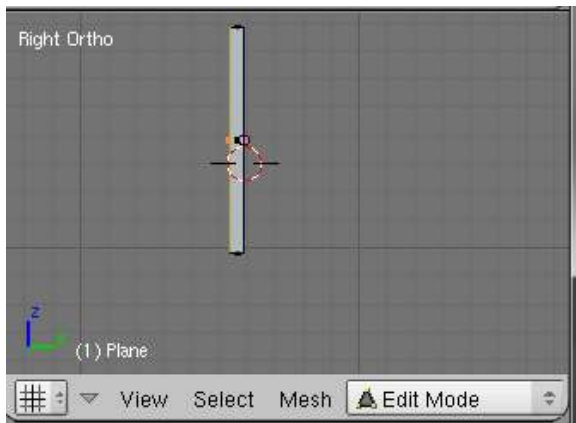


Figure 512

Switch to edit mode.

To give it some thickness, go into Side view and extrude it to 0.05.

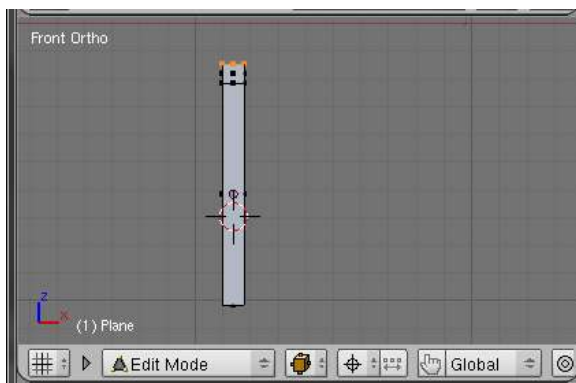


Figure 513

The pointed top is made by going back into Front view,

Selecting the top vertices and extruding them by -0.07 (along the Z axis is selected automatically),

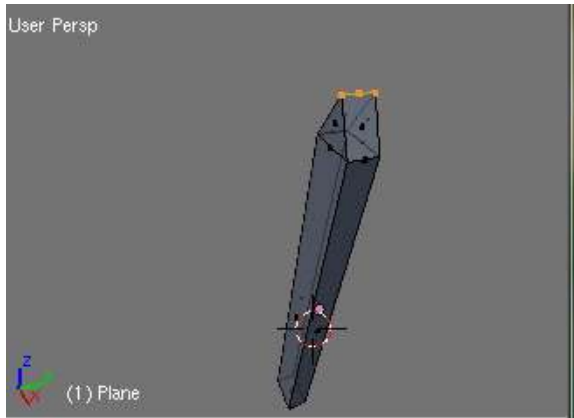



Figure 514

Scaling them to 0 along the X axis and then removing the doubled vertices ( , "Remove Doubles": it should say "Removed 2 vertices").

Noob note: If the pointed end extrusion is done before the piling is given thickness, the top doesn't move with the duplicated piling.

In Top view move the piling to the intersection of two grid lines at a suitable distance from the house.

In Front view again, check that the bottom of the piling is level with the bottom of the house.

Name and color the piling

Change the name of the object and the mesh to "Piling", create a new material for it called "Piling" and make the colour white.

62.6.2. Duplicate the Paling (Method 1)

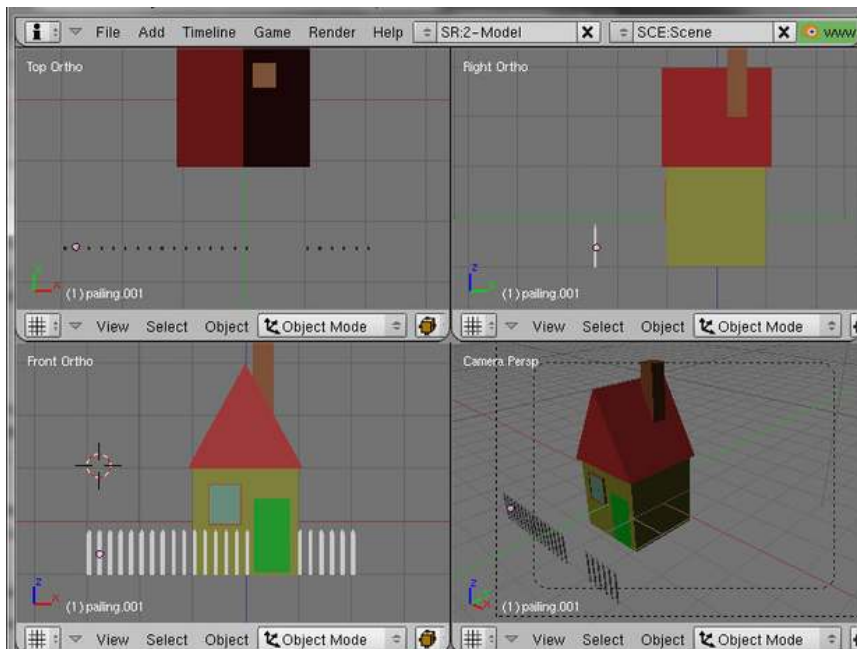


Figure 515

Be in Front view, as the flat face of the paling is along the X axis and it is easier to create the front fence first.

Change to Object mode.

Select the paling and press **[ALT] + [D]** and "0.2", **[ENTER]** to move the duplicate into position along the X axis 0.2 units away from the first one.

Noob note: we are using **[ALT] + [D]** instead of **[SHIFT] + [D]** as it creates a new instance of the paling. If you wish to make a change to the paling in edit mode, you just need to change one and all the other instances will change too.

Continue until you have the left hand half of the front fence complete, there should be a gap between the left-hand and right-hand halves for a path up to the front door. You can make this by increasing the gap at this point to 1.4 — this number will vary according to where the palings either side of the front door are placed.

62.6.3. Duplicate the Paling (Method 2)

This method uses *array modifiers* to save work.

Be in Front view, as the flat face of the paling is along the X axis and it is easier to create the front fence first.

Change to Object mode. Select the paling. Switch to the Editing context in the buttons window. Find the Modifiers panel. Click the “Add Modifier” button, and in the menu that pops up, select “Array”. You should see a second copy of the paling appear next to the first.

Look down under the heading “Relative Offset”, there are values for X, Y and Z. Change the X value to something like -2 (the negative value causes the duplicated paling to appear on the other side, away from the gap we want to leave in front of the door). Now increase the number under the popup that says “Fixed Count”, and watch the line of duplicates lengthen until it reaches the point that you want to be the corner of the fence line—a count of about 10 did it for me.

Now type `[ALT] + [D]` to make a duplicate of the row of palings, followed by `[X]` to constrain its movement to the X-axis (parallel to the row of palings). Move the duplicate row to the end of the existing row so it looks like a continuation of it. Now type `[R] [Z] -90 [ENTER]` to rotate it so it lies parallel to the side of the house. Increase the array count for this row to something like 24 so it extends all the way past the side of the house to a suitable corner point.

Again, type `[ALT] + [D]` to duplicate this second row, this time followed by `[Y]` to constrain its movement parallel to the side of the house. Move the duplicate so its looks like a continuation of the same row. Rotate it parallel to the back of the house by typing `[R] [Z] -90 [ENTER]`. Adjust its array count as necessary to get the right length.

Once more, type `[ALT] + [D]` to duplicate this third row, followed by `[X]` to constrain its movement parallel to the back of the house. Move the duplicate to the end of the row, and rotate it to be parallel to the side of the house with `[R] [Z] -90 [ENTER]`. Give it the same count you did to the array on the other side, so its length will be the same.

One last time: `[ALT] + [D]` to duplicate, `[Y]` to ensure it only moves parallel to the side of the house, move to the end of the row, rotate into position with `[R] [Z] -90 [ENTER]`. You’ll probably find it’s so long it runs into the original row of palings you started with on the other side of the front of the house; reduce its array count to something like 9 and you should end up with a nice gap in front of the door.

62.6.4. Make the horizontal bars (left side)



Figure 516

The fence needs horizontal bars to hold the palings in place: be in Front view

`[LMB]` below the middle of the left-hand fence.

Create a plane, rotating it to the vertical as necessary (**R** **X** 90 **ENTER**) and scale it to the length of the fence.

Scale it again to 0.05 along the Z axis and move it into position. (if you keep **CTRL** pressed while you move it, it will snap to the grid.)

Go into Edit mode (**TAB**)

In Side view move it so it is against the back of the fence and extrude it by -0.04 along the Y axis.

Go into Object mode (**TAB**). Give the horizontal bar the same material you previously created for the palings.

Duplicate it (**ALT** + **D**) and move it (keeping **CTRL** pressed) along the Z axis (**Z**) to make the bottom bar.

62.6.5. Make the horizontal bars (right side)

To make the bars on the right-hand part of the fence: In Front view duplicate the left hand top bar and move it along the X axis (**X**) to the top of the right-hand fence.

Scale it to the width of the right-hand fence.

Duplicate this top bar and move it down (**Z**) to become the bottom bar of the right-hand fence.

62.6.6. Put the fence around the house (Method 1 (above) cont'd)

You only need to do this if you used Method 1 (above) to duplicate the paling.

It is a bit tedious creating each paling individually so, now that you have a few ready made, you can just copy them to make the fence at the back of the house.

In Top view.

Deselect all (**A**),

Box select (**B**) the whole front fence, including the gap, Duplicate it (**ALT** + **D** again) and move it to the back.

Go into Back view (**CTRL** + **NUM1**)

Select the shorter of the two fences and box select it in the camera view; then, in the back view, move it over, next to the longer fence, till they look continuous; then duplicate the shorter fence, and move it to the left to fill in the space. (This will take a little thought, as you need to use the front/back view and the top view to line the two fences up properly.)

Delete any unneeded palings that are hanging off the edge of the back, and scale the horizontal bars in the X axis (and then move back to where it needs to go) to fit what's left.

The back fence has the horizontal bars on the outside so it must be rotated:

In **Top** view, select the whole of the back fence,

press **R** and type in 180 followed by **ENTER**.

Move the fence back into position with **CTRL** pressed to snap it to the grid.

The back fence is already selected from the rotation:

Duplicate it (**ALT** + **D**), press **ENTER** as it is not going to be moved and in **top** view,

Press **R** to rotate it and type in 90 followed by **ENTER** Press **G** to move it into position as the right hand fence and **ENTER** when it is there.

Duplicate it, rotate it and move the copy to the other side of the house.

62.7. Put the Horizontal Bars Around The House (Method 2 (above) cont'd)

You only need to do this if you used Method 2 (above) to create all the palings.

Select the two horizontal bars you created above for the front fence (make sure you don't select anything else, like the palings). Press **ALT** + **D** to make a duplicate, then **R** **Z** to rotate the duplicate about the Z-axis, type 90 for the angle and press **ENTER** to confirm the rotation. In the top view, move the duplicate (**G**) you just made so it's against the side fence. Keep scaling it along the Y-axis (**S** **Y**) and/or moving it along the Y-axis (**G** **Y**) until it's the right length and position.

Having done the side of the house, press **ALT** + **D** to make another duplicate of the horizontal bars, and **R** **Z** 90 **ENTER** to rotate it into the right alignment for the back fence. In the top view, move (**G**) it against the palings. Now keep scaling and moving it along the X-axis until it has the right length and position.

Having done the back of the house, do the other side: another duplicate of the bars, rotate 90° around the Z-axis, move against the palings, move and scale until they're the proper length in the right position.

Again, another duplicate, rotate and move/scale into position against the palings on the other side of the front of the house.

UNKNOWN TEMPLATE Clr

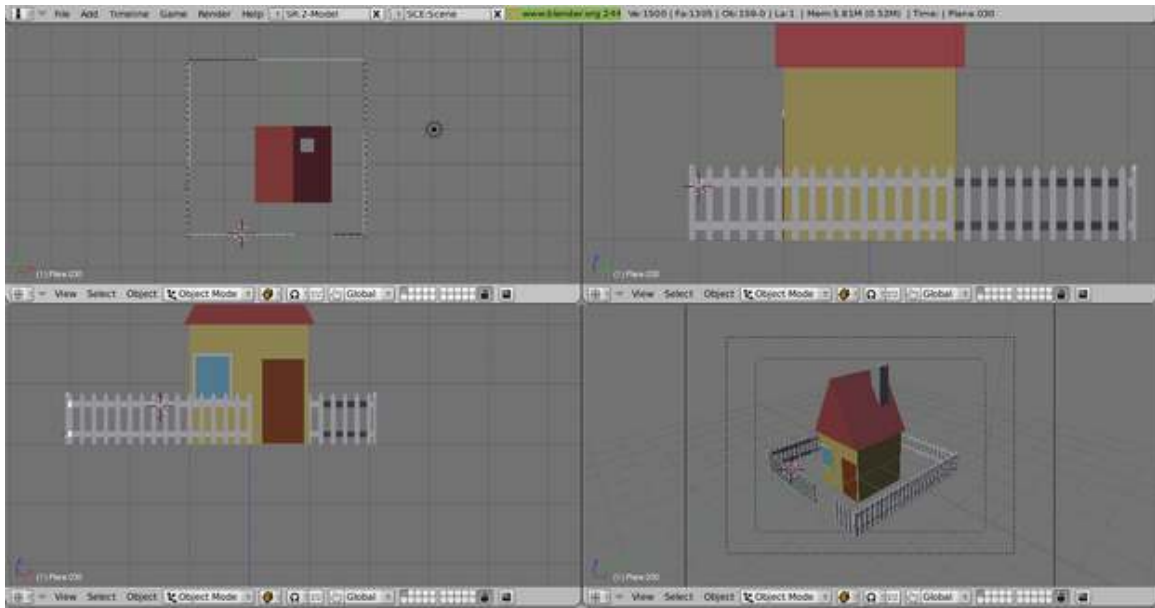


Figure 517

Your house should now look like this.

62.8. The Ground plane and a Path

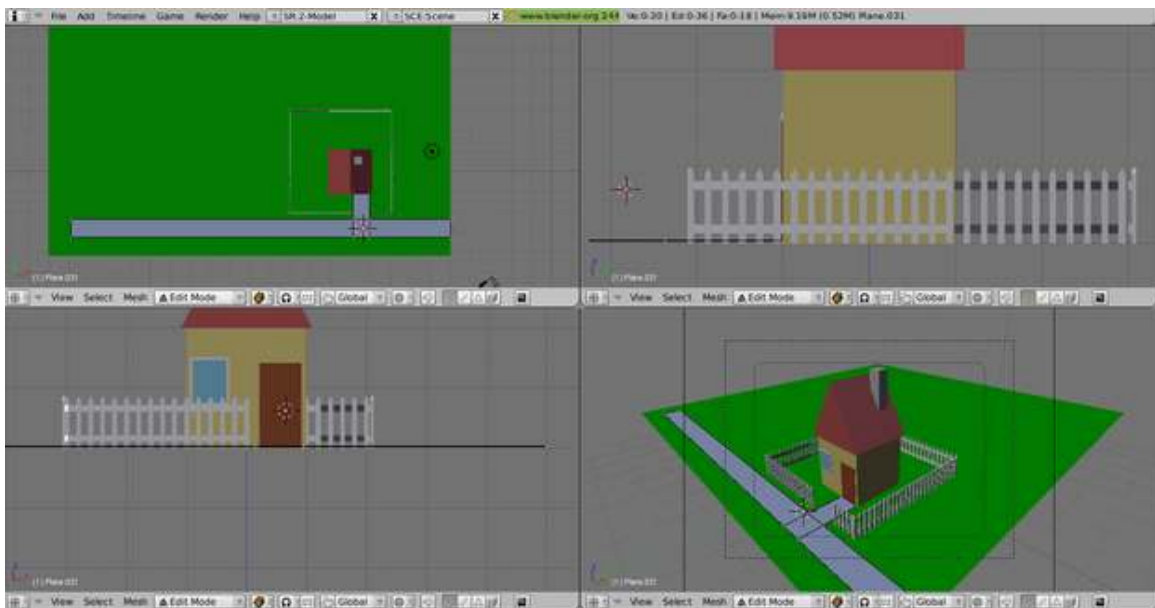



Figure 518

62.8.1. Make the ground

In Top view put the 3D cursor somewhere near the middle of the plot, (click )

Create a plane

Be in camera view (bottom right-hand window)

Scale it to extend just past the boundaries of what the camera sees (10.0 seems a good number).

Go back into Object mode and change its (object, mesh) name to "Ground",

create a new material for it, rename it "Grass" and select a suitable green from the colour selector ("00cd00" green3).

Check in Front view that it is level with the bottom of the house and fence and move it (along the Z axis) if necessary.

62.8.2. Make the path

For the path.

Go into Top view,

Create a plane,

Scale it to fit the gap in the front fence with some space on either side and position it just outside the fence.

Check, in Front view, that it lines up with the door and is level with the bottom of the house.

Extrude it along the Z axis by 0.01 to give it a little bit of thickness.

Scale it along the X axis to the same width as the door.

Be in Top view,

Extrude the top edge of the plane to the door

Then extrude the left and right edges to a suitable distance to the left and right to make the path passing the house.

62.9. Final render

Noob note: If you followed the above instructions exactly, your final render will not look as well-lit as this. Once you have done some tutorials on lighting, feel free to come back to your saved copy of this house model to improve its lighting to look more like this picture.



Figure 519

UNKNOWN TEMPLATE C1r

63. Pipe joints

Information

Applicable Blender version: 2.46

How to model pipe joints.

63.1. T joint

Start in top view, and delete the cube.



Figure 520 **Figure 1:** Modeling a simple T joint.

- Add a cylinder object (*Add->Mesh->Cylinder*)
- 16 Vertices, No Caps, Depth 4 (For recent versions of Blender create the cylinder then press F6 to get these options)
- Switch to front view (*View->Front or NUM1*).

The X axis now points to the right. We will work along the X axis from now on.

- Change to Edit mode (TAB).
- Create a Loop Cut in the middle of the cylinder (**Ctrl-R**).

Make sure that the 3D cursor is placed in the middle of the cylinder.

- Switch to right view (**NUM3**)
- Box select (**BKEY**) the 7 vertices at the front (Tip: tick the *Occlude background geometry* button)
- Switch back to front view (**NUM1**)
- *Pivot -> 3D Cursor* (**.KEY**)
- Scale with the X axis (**S-X**), about 1.4142 (square root of 2).
- Rip the mesh with **V**.
- Depending on which half the vertices are now selected you will need to rotate by -45 or +45.
- Deselect all (**AKEY**), then box select the lower half of the opening
- Rotate the selected vertices by 45 degrees. Now the pipe should be straight again thanks to the rotations.
- Box select the vertices of the gap and extrude "edges" along the X axis.
- *Pivot -> Median Point* (**CTRL+,KEY**)
- To flatten the front: scale at the X axis by factor 0. (**S-X, 0**).
- Use the *Subsurf* modifier in combination with *Set Smooth*, *LoopCuts*, and *Crease* to get a nice transition.

63.2. 6 cylinder joint



Figure 521 **Figure 2:** 6 cylinder joint

Be in Object mode (have nothing else selected)

Add a cube.

Switch to Edit mode

Select all, then Subdivide the cube once (**W**, **1**) (**1**).

Select the 12 vertices in the middle of the cube's edges.

Scale by factor 1.4142 (press **S** key and type in the factor with the keypad) (2).

Select all vertices, and Extrude "Individual Faces" (This can be done either on the Tool Shelf under "Add" or you can type **Space** and search for Extrude Individual Faces and Move.) (3).

Select all vertices, and *Remove Doubles*.

Select the center vertex of each tube, and delete them to open the ends

Select the original vertices in the middle of the object and delete them (5).

Clean your mesh as described above.

63.3. 3 cylinder joint



Figure 522 **Figure 3:** 3 cylinder joint

This is very similar to above, only the selection for scaling is different.

- Select the middle vertices at the edges where the cylinders shall come from. Additionally select the middle vertices at the faces that shall be kept free. You have to select $12+3=15$ vertices.
- Scale your selection by 1.4142.
- Select the faces that shall be extruded.
- *Extrude->Individual Faces (5)*
- Select all, then *Remove Doubles*.

- Delete the inner vertices (**6, 7**).
- Select 3 vertices (which were middle of faces that were kept free) and 1 corner vertex in middle of the free faces (**8**).
- Make sure your 3D cursor is at the origin (and that you added the cube at the origin)

and switch to 3D Cursor Pivot (**.KEY** in v2.49b), then scale (**SKEY**), and type in 0.7071 (inverse of square root of 2, or Sin(45 degrees) or Cos(45 degrees)) and **ENTER** (**9**).

- Make sure you switch back to Median Point Pivot (**SHIFT+,KEY**) to straighten out pipe ends (**10**).

Clean your mesh as described above.

If your joint looks odd, it might be because of some edges in the mesh. Delete those by selecting it and delete (**XKEY**) edges (not vertices).

63.4. T joint with smaller diameter

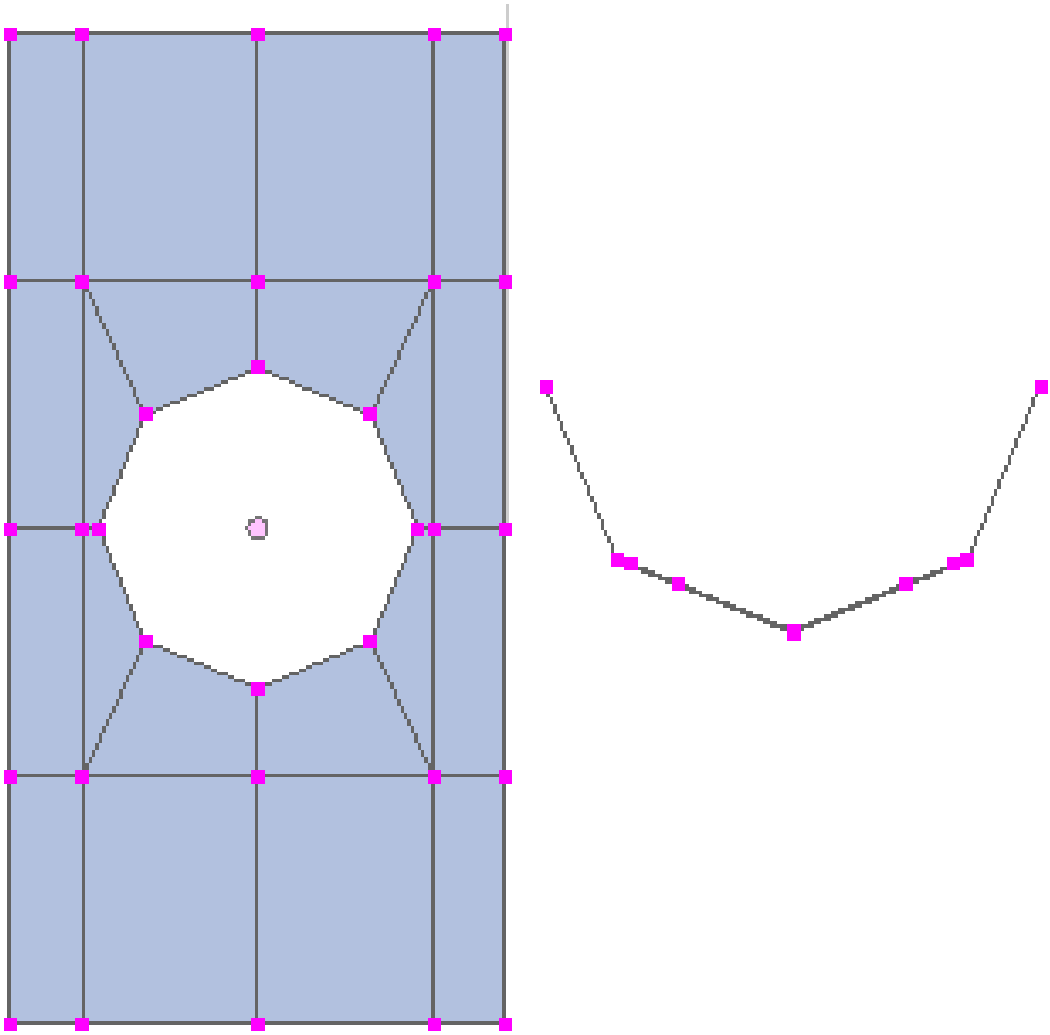


Figure 523 Figure 4a: Basic structure for a T joint with a smaller diameter

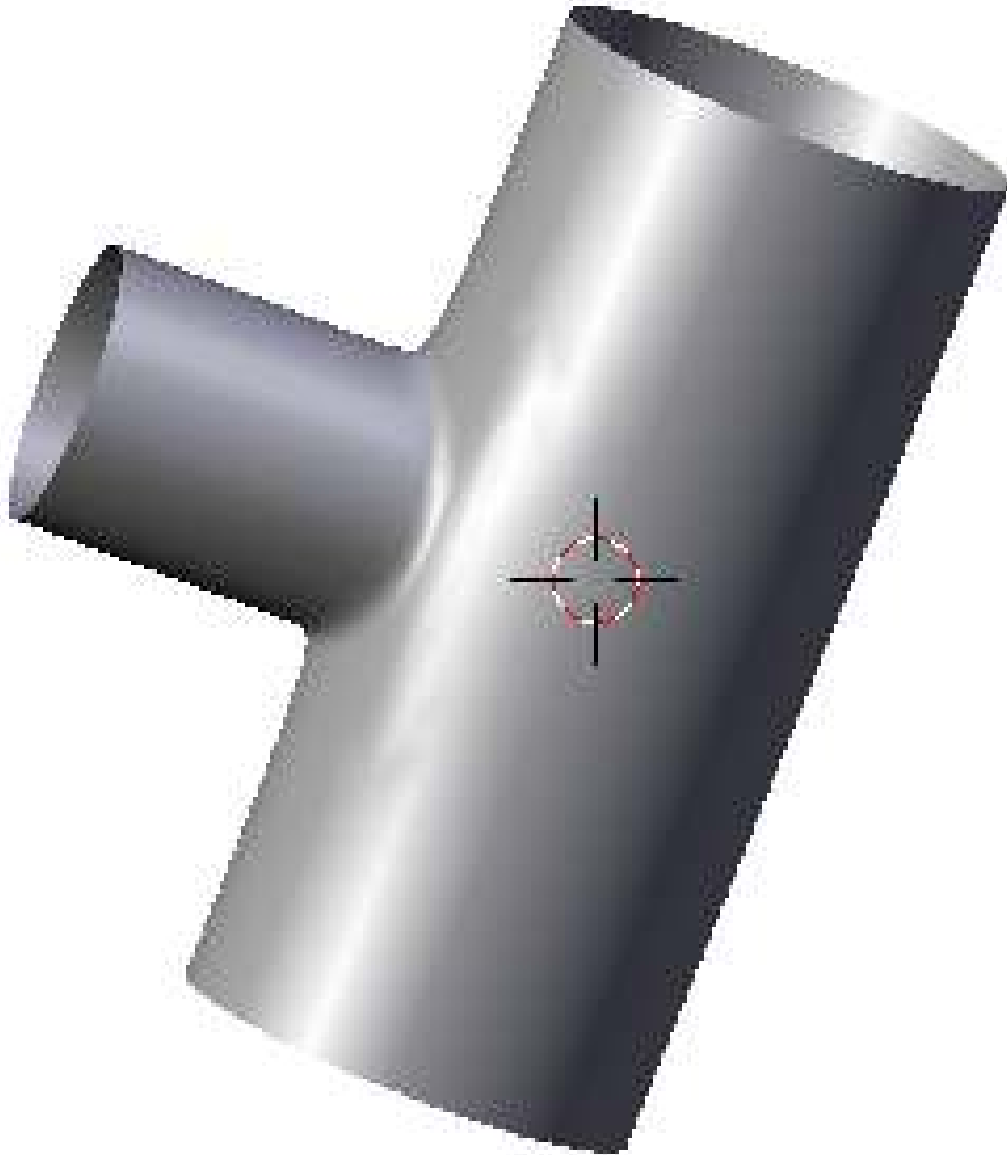


Figure 524 Figure 4b: The result

We're going to create a structure like in **Fig. 4a** and extrude the inner circle. To do that we place a circle with *Retopo* (Manual¹) on the cylinder and join both objects afterwards.

- *Add->Mesh->Cylinder*
 - 8 vertices (of course you can use more vertices if you like)
 - *No Caps*
 - *Depth 4*

¹ http://wiki.blender.org/index.php/Doc:Manual/Modelling/Meshes/Remaking_the_Topology

- **Noob note:** *More vertices in your cylinder will mean more intuitive work and figuring [more loop cuts, face making, etc.] later on when retopo-ing. I suggest following the tutorial first, then try to do a more advanced model with more faces / vertices when you understand the tool. If you have trouble [as I did] getting really nice geometry, do a search for some tutorial videos². There are many really helpful ones.*
- Add->Mesh->Circle (**Noob note:** *To make sure you create these as separate objects, be in Object Mode (not Edit Mode).)*
 - 8 vertices
 - No fill
- Rotate the circle by 90° at the Y axis (**R->Y->90**).
- Move the circle in front of the cylinder(**G->X->2**).
- Switch to side view (**Num-3**).
- Scale the circle to the diameter of the smaller cylinder.

(**Noob note:** *I felt this was a little unclear. What the author means is, scale the circle down until it is the size of the diameter of the smaller cylinder you are going to create from it later by extrusion. You would be wise to scale it so it fits (straight backwards from view) inside the two closest faces of the cylinder as per **Fig. 4a** if you wish to follow the tutorial.)*

- Switch to edit mode, select all vertices of the circle.

(**Noob note:** *To recap: you should be in edit mode for the circle, with the all of it selected, with the cylinder object directly behind it in from your viewpoint in orthographic mode.)* Make sure that you don't work in Wireframe view.

- Click on Retopo->Retopo All in the Mesh panel of the Editing buttons.

(**Noob note:** *What Retopo does [in this case]; is project the selected topography of an object in edit mode directly backwards from your selected view point onto the topography of another [seperate] object. So you must be lined up view-wise for this to work. I've also found sometimes I must press the **Enter** key for the **Retopo All** to take effect even if I've selected it with the **LMB** .)*

If the vertices of the circle are not adjacent to the cylinder but have jumped to wrong places undo the last step (**Ctrl-Z**), deselect and reselect all vertices and try again. This does help (strangely). Retopo works with limited accuracy, if the vertices don't fit perfectly you have to move them by hand.

- Change to object mode, select additionally the cylinder and join both objects (**Ctrl-J**).

Now you work best in Wireframe view.

- Add three loop cuts (**Ctrl-R->3**)
- Delete the middle vertex.
- Connect the free vertices (**F**).
- Select the circle.
- Extrude and clean up.

Noob Note -: I couldn't follow this last pipe tutorial with blender 2.6, so I made a video on how to do it once i worked it out. Thought i would put link to it in case anyone else needed help. (hope that's OK)

<http://www.youtube.com/watch?v=uGm696IyLEU&feature=plcp&context=C4a1ad76VAvjVQa1PpcFOwQ4POxHxgJsc9jJcxe85fo%3D>

2 http://video.google.com/videosearch?sourceid=navclient&rlz=1T4GGLR_enUS262US264&q=blender%20retopo%20tutorial&um=1&ie=UTF-8&sa=N&hl=en&tab=vw#

64. Aligning Vertices with a Guide Image

This tutorial is about using guide images to place vertices in their proper places in 3D space. The second tutorial is on how to take good reference pictures. This tutorial assumes that you have completed all previous tutorials.

This tutorial describes the use of the *background image* feature of Blender to assist in creating models of 3D objects. The background image provides a reference for the dimensions of the object, similar to the way the floor plan and elevation views of a house provide dimension information for the actual house. Guide images are not rendered, and may be removed after the model is completed, or they may be retained as part of the internal documentation of the model. Background images are not generally useful for other "image" purposes such as materials, textures, and actual background images in a scene, just as a floor plan is not actually visible after a house is constructed.

64.1. Background: orthographic projection

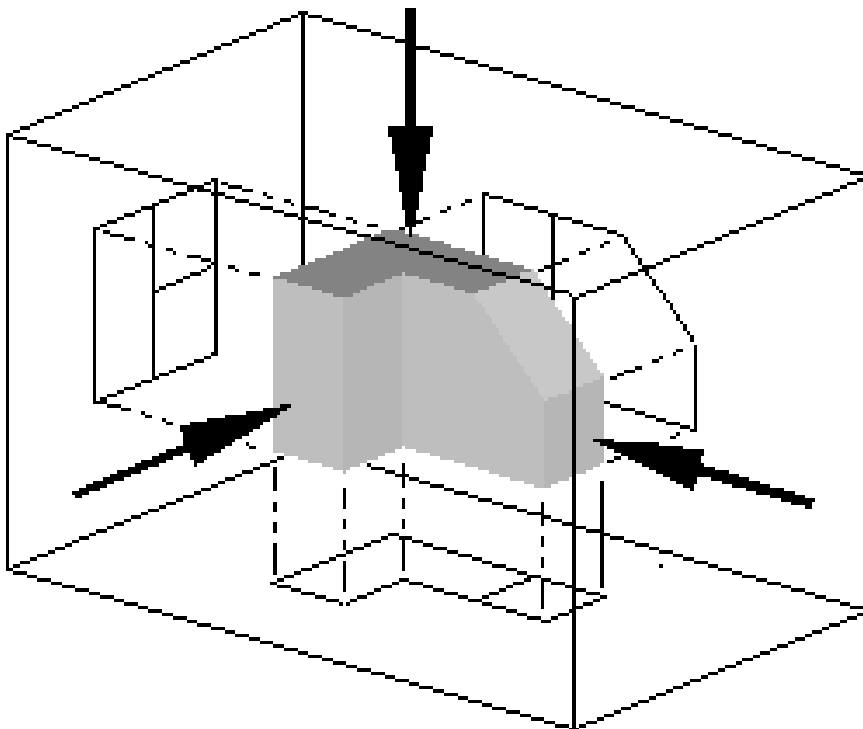


Figure 525 orthographic projections in six directions

orthographic projection¹ is a technique used by architects and engineers to describe a three-dimensional object by the use of several two-dimensional images, or "projections." In architecture, the projection as seen from above is called the "plan view" or "floor plan," the projection as seen from the front is called the "front elevation," and the projection as seen from the right side is called the "right elevation." Orthographic projections are intended to assist builders in creating an actual 3D object.

A Blender user also wants to create a 3D object, in this case a model. Blender provides a way to use a set of one or more orthographic projections as a guide for object creation. Because the images are conceptually "behind" the object from the appropriate point of view, Blender calls these 2D images *background images*.

Each background image is located "at infinity" in an orthographic view, and there may be one image in each direction: back, front, top, bottom, left, and right. we can use one or more background images to assist our modeling effort.

For example, to model a house, we can put the "floor plan" on the bottom, and build the house above it. We put the front elevation image on the back background, and build the house in front of it. We put the right elevation image on the left background, and build the house to the right of it. The background images can be diagrams such as floor plans, or they can be photographs of an object taken from sufficiently far away to provide undistorted dimensional information.

Reasonably enough, an orthographic projection is only useful in Blender when in orthographic mode. Blender enforces this to prevent you from making mistakes: the orthographic image is only displayed in the background when you are in the orthographic view mode.

¹ <http://en.wikipedia.org/wiki/Multiview%20orthographic%20projection>

64.2. Making a Simple Pyramid

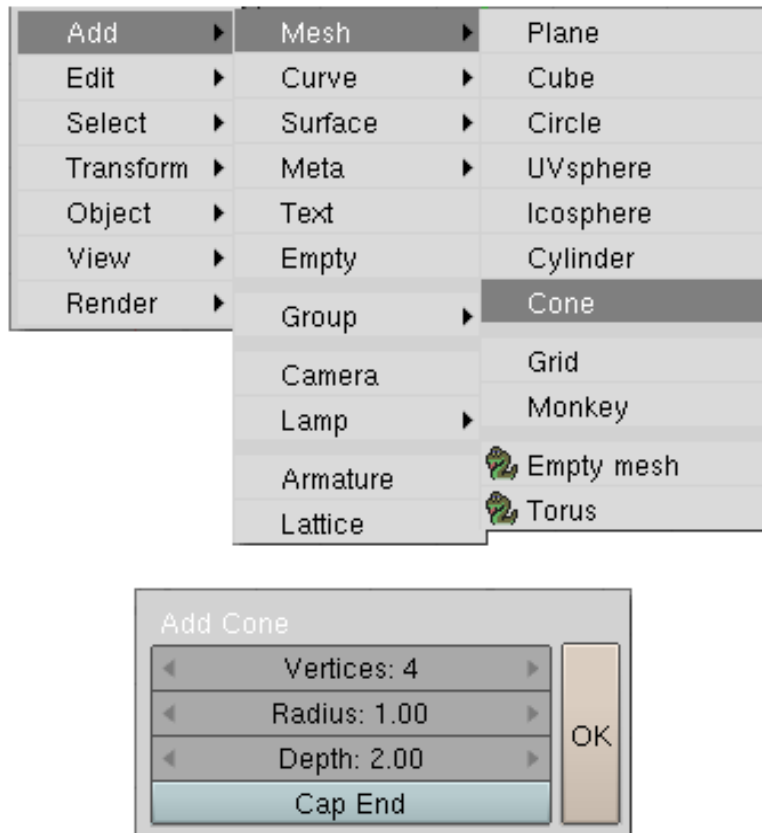


Figure 526

First we are going to create a pyramid the easy way. Then we are going to show how to use different viewpoints and images as a guide to place vertices correctly in 3D space.

Get rid of the default cube. Press **[SPACE]** and select Add→Mesh→Cone. Set the number of vertices to 4, and unselect Cap End. Click OK. There's your pyramid.

Note: Do not render this looking up from the bottom, as it will appear invisible, as the interior faces of models are ignored by most rendering engines

64.2.1. Save the top vertex for later steps

Enter Edit mode (**[TAB]**). First, unselect all the vertices by pressing **[A]** . Next, select the bottom four vertices of the pyramid and delete them with **[DEL]** or **[X]** . The only vertex left will be the vertex which makes the tip of the pyramid. This will be used later.

64.3. Using the guide images

Now that we have the pyramid the easy way, let's learn how to use guide images as references to build models.

64.3.1. Window Layout

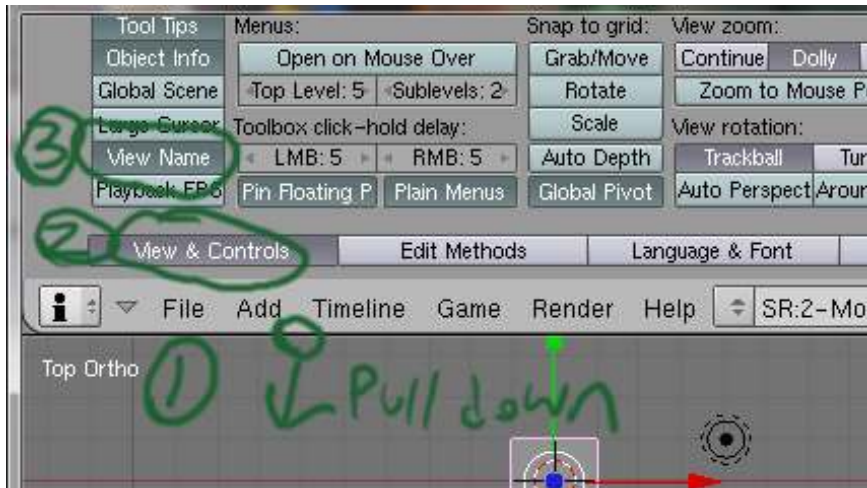


Figure 527

To make things easier, open a user preferences window² and in the "View & Controls" sub-menu, turn on the "View Name" option. The window names will be used for reference inside the rest of this tutorial.

Split the Main 3D view window in to 4 windows.

Reminder: to split windows, move the mouse to the border of the view, when the cursor transforms into arrow, right-click and choose "Split Area". (Explained in the guide: Noob to Pro/Blender Windowing System³.)

Change the point of view in each window so that they end up like this:

NUM1
NUM3

NUM7
NUM0

And if you click on View, you can see that these windows are respectively:

² Chapter 14 on page 69

³ Chapter 13.3 on page 59

Front
Side (Right)

Top
Camera

By knowing which view you are looking at you may find that you quickly get the idea of what you are doing and can proceed somewhat intuitively in this section on your own without following all of the step by step instructions.

64.3.2. Guide images

Now, we need some images. These come from a source outside of blender. For this tutorial, we need a floor plan of our pyramid, a front elevation view, and a side elevation view. The floor plan of our pyramid is a square, and the front elevation view is a triangle, as is the side elevation. We will cheat and use the same source image for both the front and side elevation. We can make a picture of a white square and of a white triangle in the GIMP, Paint :) , or some other image editor. or we can find appropriate images somewhere.

Method 1 download and use

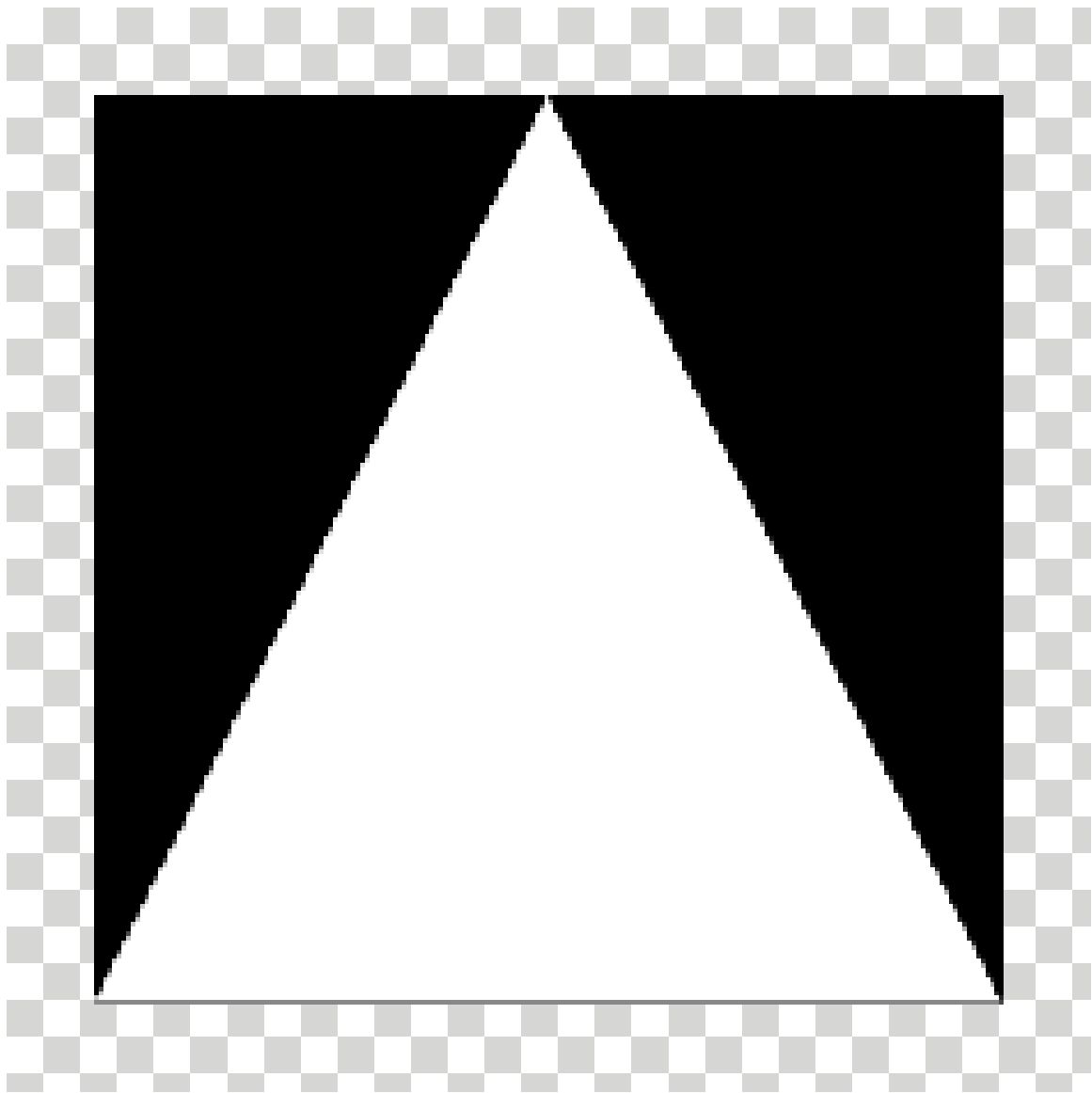


Figure 528 Save the triangle with a black background

Download the black and white triangle image on the right of the screen and use that. This image is, minus the checker pattern at the border, is 198 x 198. (click it once to get the larger version, right click on the larger one, and save)

Method 2 roll your own**Figure 529** The square and triangle relationship

Noob note Make sure that the *drawing* of the square is *square* and not just rectangular. Make the triangle the same width and height as the square. Make sure the apex of the triangle is directly above the midpoint of its baseline.

Specific instructions for Photoshop

Make a square selection of "n by n" size, remember the value of "n". Fill it with white color and save. To create a triangle of needed properties make a rectangular selection of same (n by n) size, on a new layer, click **RMB** on your document, choose "Transform selection" option in the pop-up menu. Once you are in "Transform selection" mode, right-click the blank image again. This time the pop-up menu would be different. Choose "Perspective" from it, and with **LMB** drag one of the two top vertices toward the other. Once the vertices meet (in the top-center of the image), exit the transformation mode, and fill the resulting triangular selection with white.

Save the files to a place that is easy to access. Blender only supports the TGA, PNG, and JPG image formats.

Specific instructions for Gimp

Turn on the grid (View->Show Grid, View->Snap to Grid), use the rectangle select with a fixed aspect ratio of 1:1 (in the tool options panel) to select a square that you can flood fill. For the triangle, use the node tool to draw a triangular path, convert to selection (Select->From Path) and fill it. Or you could just use Inkscape...

Save the files to a place that is easy to access. Blender only supports the TGA, PNG, and JPG image formats.

64.3.3. Background Images

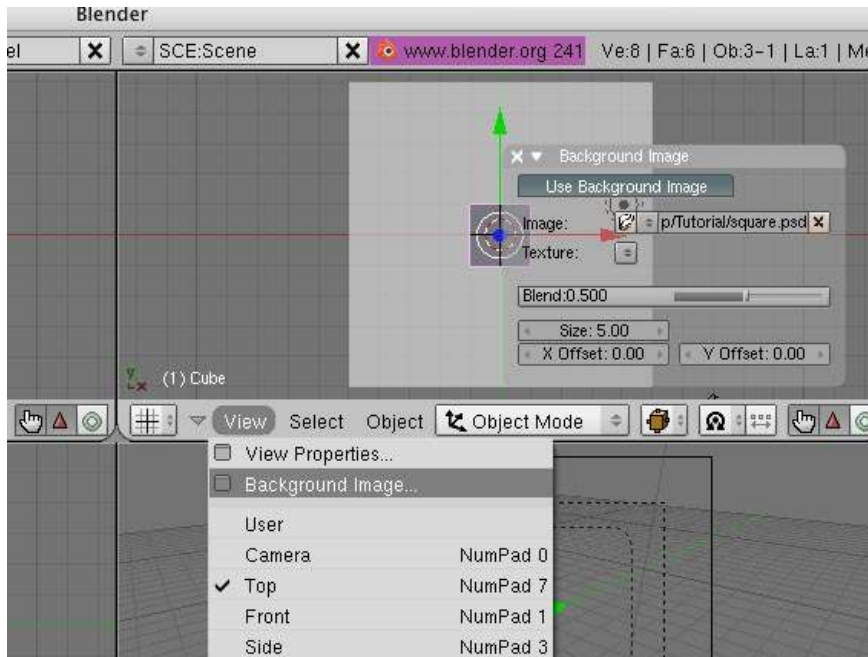


Figure 530

Load the white square (the floor plan) into the **top (NUM7)** window by going to the 3D view window and pressing *View* → *Background image* → (click on the icon of a file). Find your file and click "select image". Since you are placing the background while looking at the view from the top, the background is placed infinitely far away at the bottom of the model, which is where we want it.

I have a more recent version of blender, and this method does not work. when I hit "view", background image is not an option. I looked for a solution, and found this video: "http://www.youtube.com/watch?v=j14Irv_5UgI&feature=related"⁴

Load the white triangle (the front elevation view) into the **front (NUM1)** view window: this places it infinitely far "behind" the model. Similarly, load the right elevation (coincidentally, the same triangle image in this special case) from the right side view **side (right or NUM3)** window, to place it infinitely far to the left.

If necessary, zoom out so that you can see the whole background image in each view.

Now you have now placed your guide images for making your pyramid.

Note: for each view, if you can not see the picture, switch to Ortho view by hitting NUM5 You won't see it in top view till you do this. Furthermore, the entire idea of using guide images is based on using the orthographic projection, because in the orthographic projection, points on the same line normal to the viewing plane appear at the same point on the view.

⁴ http://www.youtube.com/watch?v=j14Irv_5UgI&feature=related

64.3.4. Side One

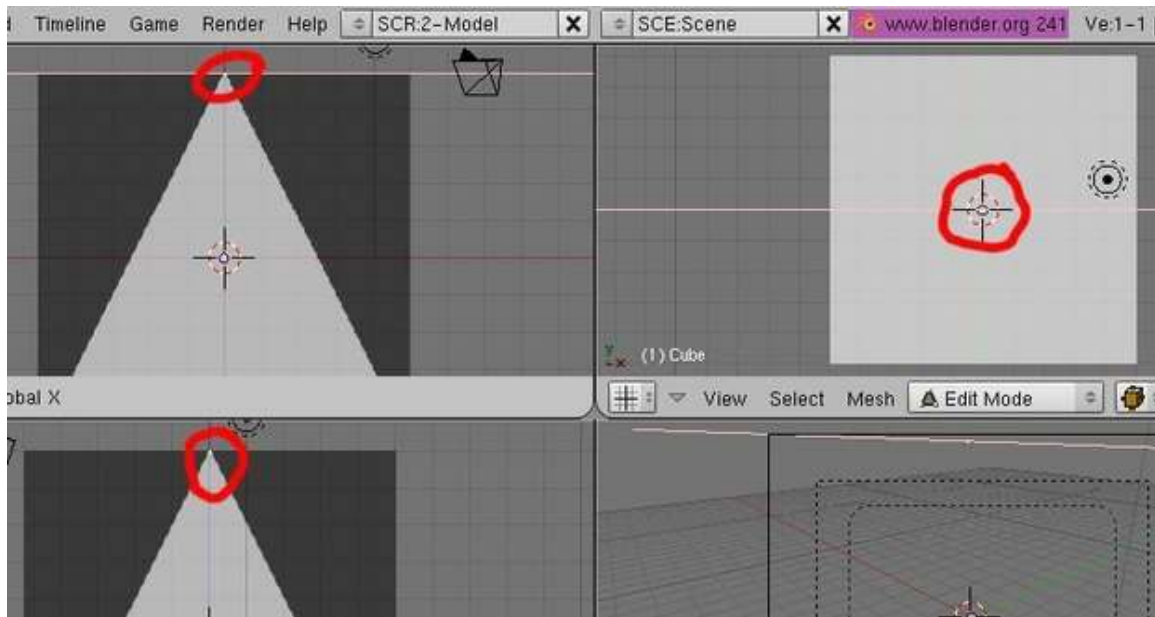


Figure 531

The vertex that is left will be the topmost point of the pyramid. Use the **GKEY** to move the vertex around. To get it in the right spot, line it up at the top most point in the **front (NUM1)** and **side (right or NUM3)** windows. If you look in the **top (NUM7)** window the vertex should appear to be in the center. Make sure to keep the vertex highlighted for the next step.

Note: If you deleted all of your mesh objects in Object Mode (and remaining vertice), then you'll be stuck in Object Mode. You can however, add a new mesh object as a base, or an empty mesh via a python script (*Add → Mesh → Empty Mesh*) [Blender 2.49]. Once an empty mesh object has been created, select it (right click on it or **AKEY**), and then change to the Edit Mode. Now you can begin to create individual vertices with **CTRL + LMB**.

Note: it might be helpful at some point to zoom in and use the X,Y and Z movement restriction)

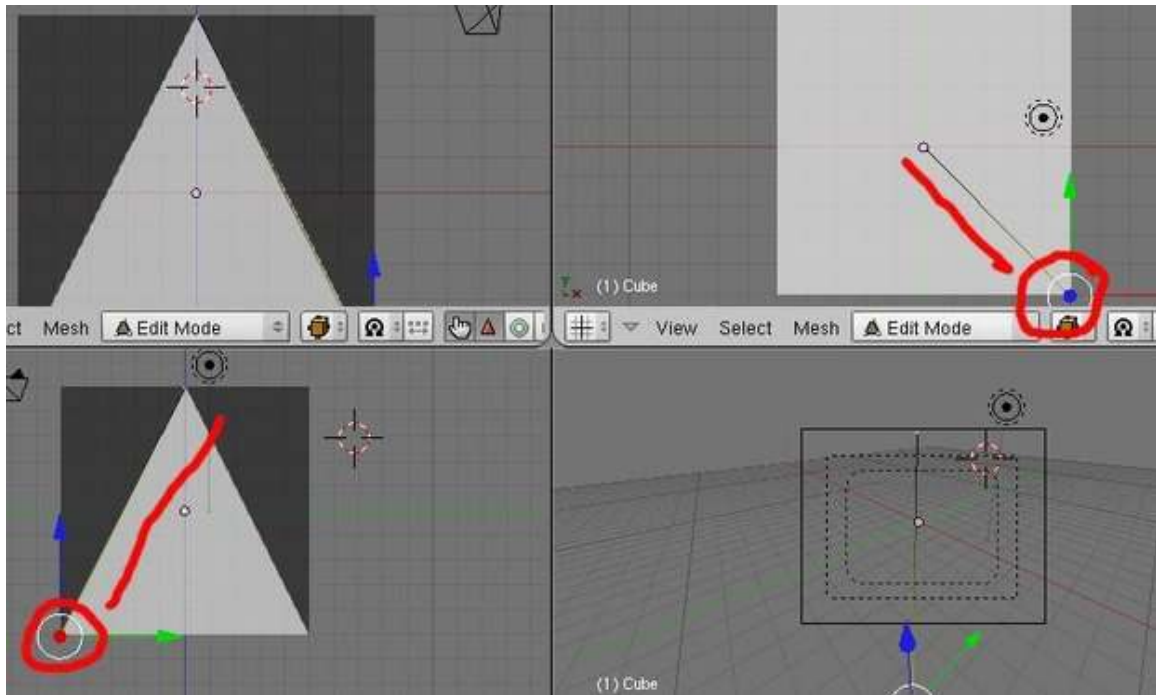


Figure 532

Now, we are going to place more vertices in the scene. Since we want to have the new vertices connected to the first one, we will make sure the first vertex is selected and hold down **CTRL** and click **LMB** to create a new one.

In the **side (right or NUM3)** window, place a vertex on the lower left edge of the triangle by holding **CTRL** and clicking there. This should create a line between your 2 points. If needed, use **GKEY** to line it up in that window and also at the lower right point in the **top (NUM7)** window.

Noob Note: if you wind up in the middle of square (top view) then move your new vertex around with **GKEY** till it's in the right place in the front, side, and top views

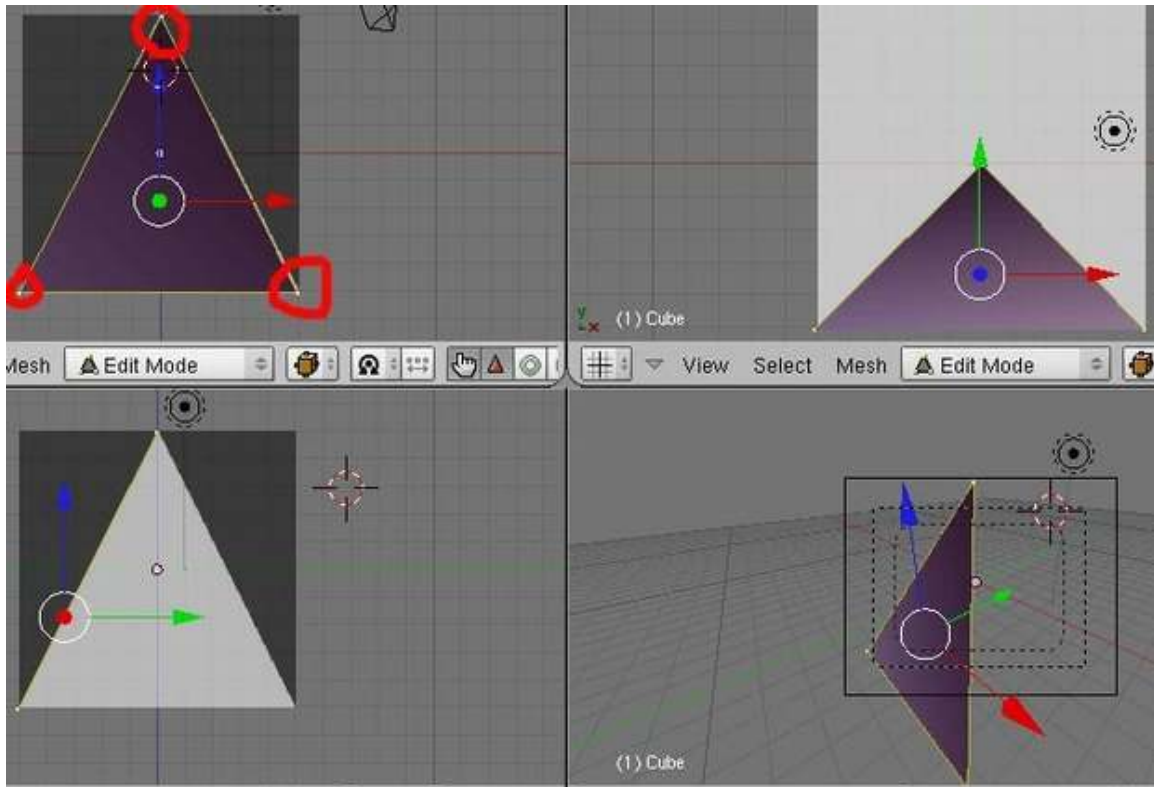


Figure 533

Next, with only one of the vertices selected, in the **front (NUM1)** window place a vertex on the lower left edge of the triangle using the same method. Line it up in that window and at the lower left point in the **top (NUM7)** window.

Now, select your 3 vertices (use **AKEY**) in the **front (NUM1)** window and press the **FKEY**. You should see a triangle appear.

64.3.5. Side Two

Press **AKEY** to deselect all vertices and select the vertex at the top of the pyramid again. Repeat the process from Side One to make the next side. Place a vertex in each of the other corners of the square using **CTRL** and **LMB**. Line them up in two different views with the corners of the triangle and square. Make sure only the top vertex and the two new vertices are selected and press **FKEY**. This will fill in a face opposite of the first face in the pyramid.

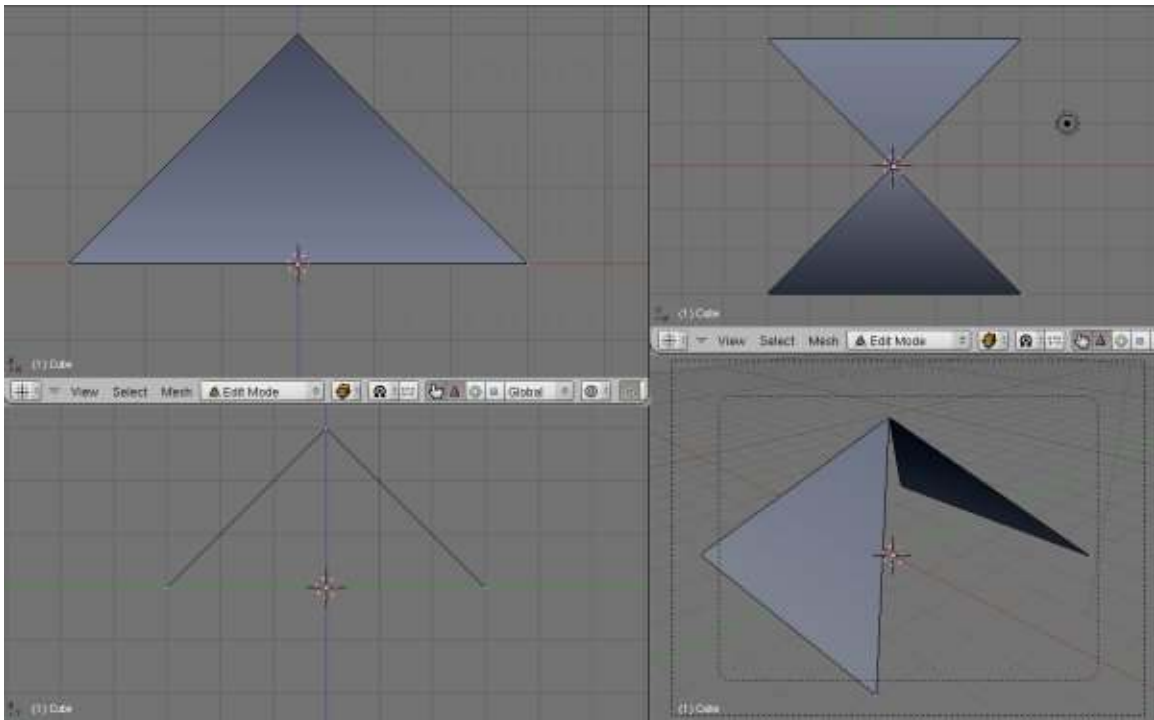


Figure 534

64.3.6. Sides Three and Four

To fill in the other two sides, select the top vertex and the two corner vertices on a side where there is no face yet. Again, use **FKEY** to fill in a face. Repeat this for the last side to have all four sides created.

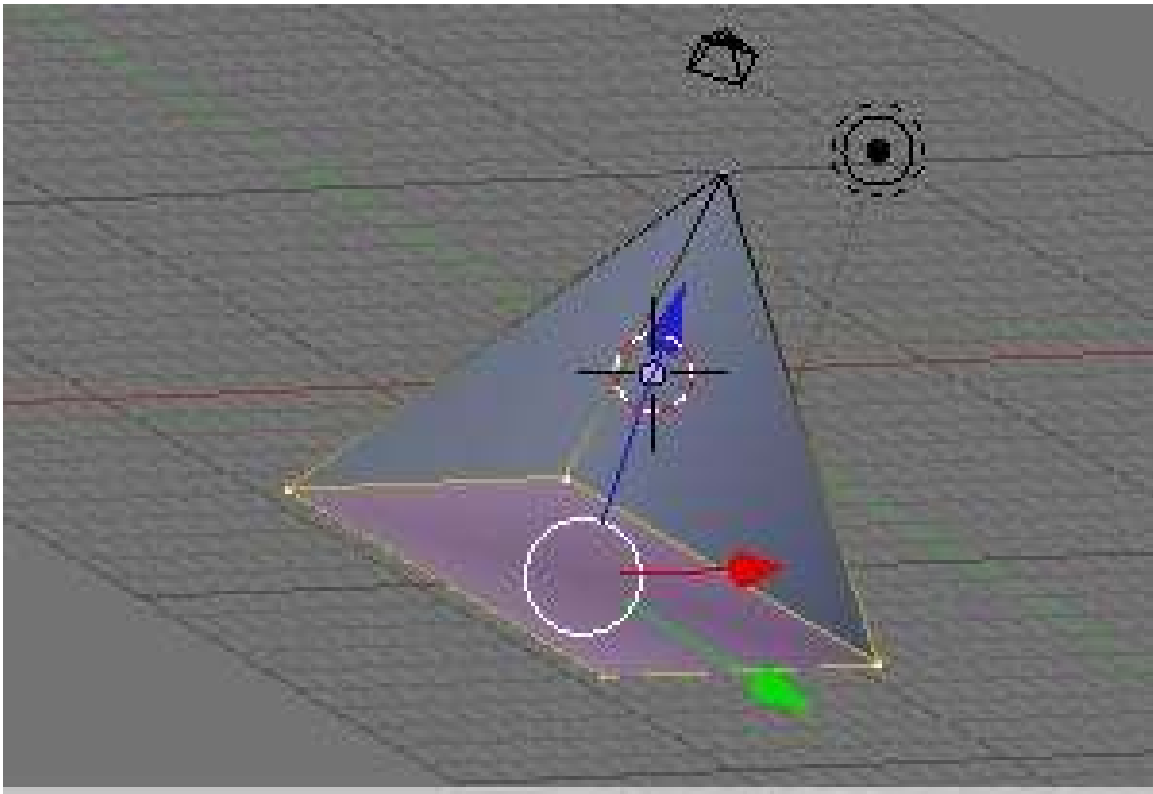
64.3.7. Bottom

Figure 535 selecting all 4 corners

In the **top (NUM7)** window, select all four corners and make a face. You should have a solid pyramid! Now, select all 4 faces, hit **XKEY**, and choose "faces".

UNKNOWN TEMPLATE Clr

65. Modeling a Fox from Guide Images

This tutorial assumes that you have completed all previous tutorials.

65.1. Method 1

65.1.1. Get the pictures of the model

Tip

The images here do not line up. Some need to be rotated and others do not match in size. They will be kept so that you can get the "real feel" for this project.

If you have a puppy and a digital camera, take three pictures of the cute little rascal and upload them. If you don't have a puppy, any object or small animal will do. Ideally, the photos will be looking straight down at the top of the puppy, a side view, and a front view. It's important that the puppy be in the same pose in all three photos! Or at least close to the same pose...we all know puppies don't stand still very long.

You could use two mirrors. One is placed next to the puppy at 45 degrees to the camera and 45 degrees to the puppy. Another is placed above the puppy, also at 45 degrees to the camera and 45 degrees to the puppy. This produces three images, one of the puppy (**front \ NUM1**), one of its reflection seen 90 degrees to the right (**side \ right \ NUM3**), and one of its reflection seen from overhead (**top \ NUM7**). Take the photo from a long distance away with a zoom lens to get close to an orthographic projection¹.

Or how about pictures of a toy wolf taken from 6 view points?:

- Left view²
- Right view³
- Front view⁴
- Back view⁵
- Bottom view⁶
- Top view⁷

1 <http://en.wikipedia.org/wiki/Orthographic>
2 <http://img51.imageshack.us/my.php?image=10011325wu.jpg>
3 <http://img162.imageshack.us/my.php?image=10011337aj.jpg>
4 <http://img107.imageshack.us/my.php?image=10011347fw.jpg>
5 <http://img107.imageshack.us/my.php?image=10011350gu.jpg>
6 <http://img162.imageshack.us/my.php?image=10011363q1.jpg>
7 <http://img119.imageshack.us/my.php?image=10011375bc.jpg>

Using your favorite image editor, such as PhotoShop or the GIMP⁸ (see detailed GIMP instructions below: Detailed steps to align images using GIMP⁹), down-scale the images need to a reasonable size (I made mine 512x384), and then match them to each other. To match them, draw construction lines (pulled from the rulers above and to the left) on the left view for example to pick out key features. I picked the tail, the front of the back foot, eye level, tip of the ear, and the front of the nose:

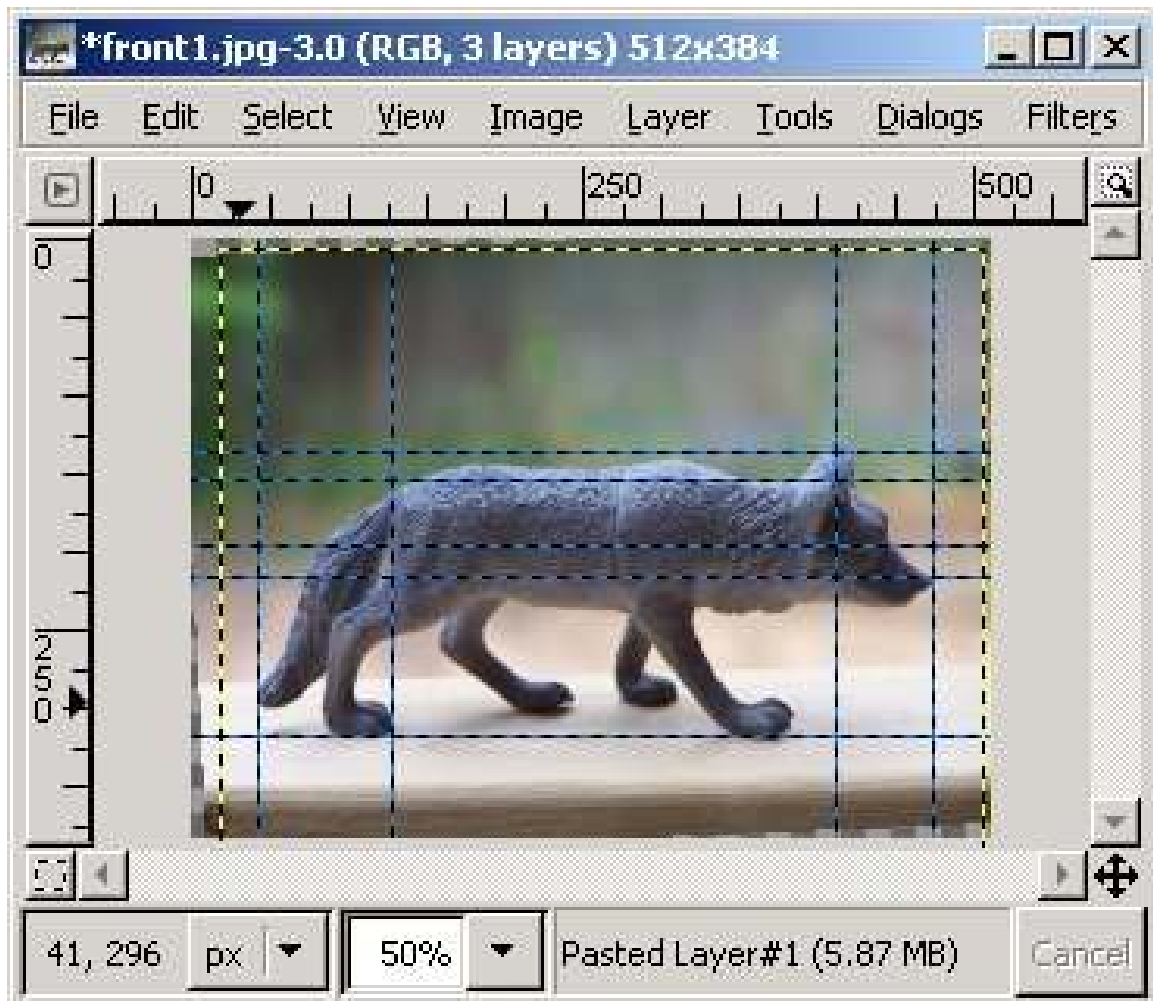


Figure 536

I found when I picked out these features that this first image needed to be rotated slightly. That completed, I proceeded to scale, rotate and shift the other two views (top and front) until they matched fairly well as layers on top:

8 <http://en.wikibooks.org/wiki/GIMP>

9 Chapter 65.4 on page 584

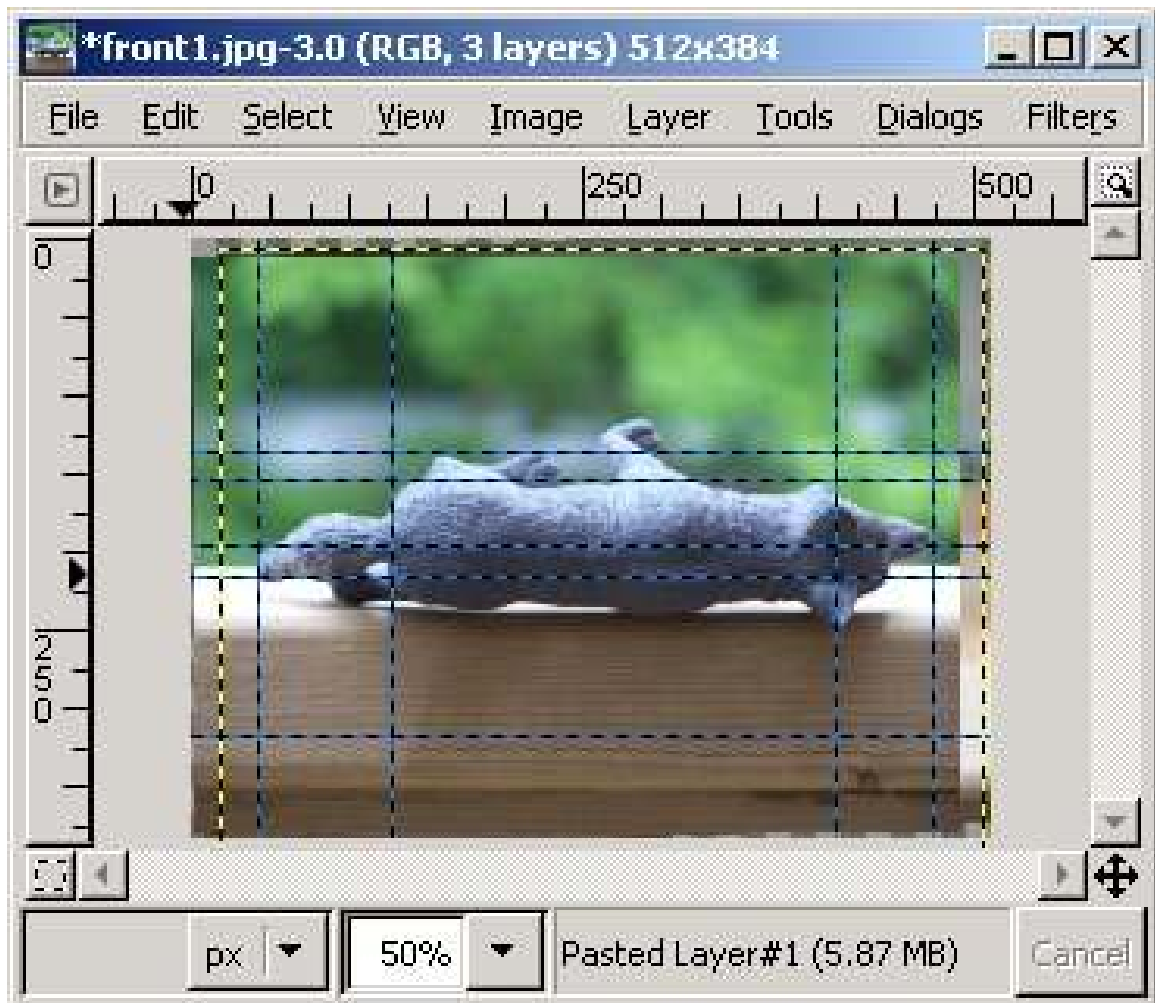


Figure 537

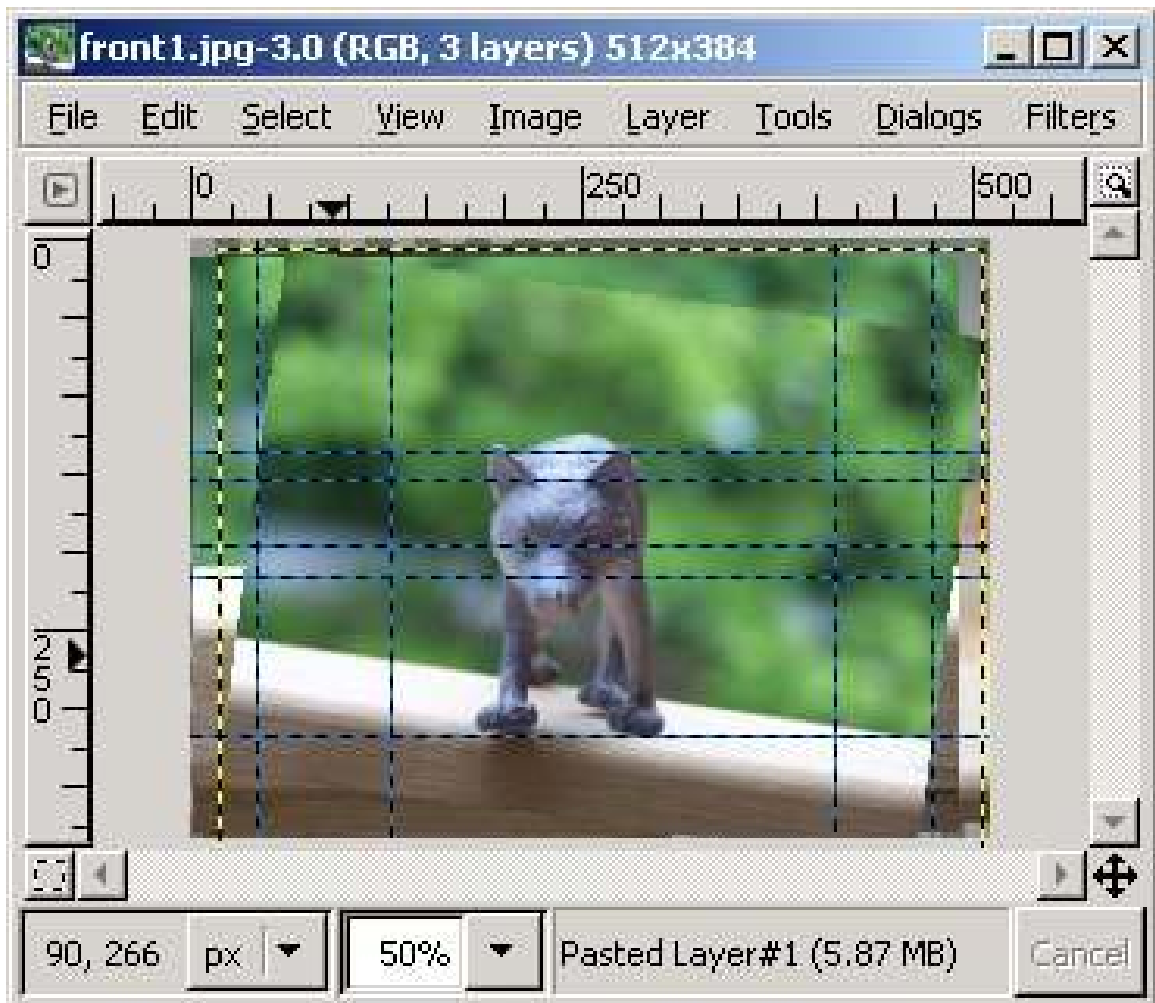


Figure 538

Once I had the proper results I saved the resulting images, and these are the ones we will use in Blender.

The results are the files you'll need for Step Two:



Figure 539



Figure 540



Figure 541

Just right-click and save them some place where you can find them to load them into Blender for Step Two. You may notice the photos aren't perfect, but we'll use them just to show how you should deal with your real photos. When you are creating your own pictures to import, note parallax¹⁰. In this example, parallax is present, and we'll attempt to compensate.

65.1.2. Get the Picture into Blender

Getting the image into blender is the easy part. The more difficult part will be creating the mesh, but first things first. Create a new file (*File* → *New*) to see the familiar default objects. Don't bother deleting the cube, we'll end up using it in the tutorial. Just as was done in the "Making A Pyramid"¹¹ section, split the 3D Viewer into four views and assign them the following view ports. Note that the arrangement is different than that used in the previous tutorial.

NUM7
NUM1

NUM3
NUM0

¹⁰ <http://en.wikipedia.org/wiki/Parallax>

¹¹ Chapter 64.3 on page 556

Each window will show you different XYZ coordinates with the **NUM0** view being what your object will look like to the camera object.

Now that the screen is set up, let's load the images. In the top left viewport (XY), click *View* on the viewport's header (the menu at the bottom). From the menu provided, select *Background Image ...*. A small window will appear.

Tip

If you have a later version of Blender and the *Background Image ...* option is missing from the View menu, try going to View > Properties. A new toolbar will open to the right of the viewport and by scrolling through you should see *Background Images*. (Blender 2.58 user)

Click the *Use Background Image* button and several more buttons will appear. Now click on the folder icon located to the right of *Image:*. A new full-viewport window will appear. Explore this window a bit and end up selecting the image file of the wolf from the top view. What you should get is the picture of the toy wolf from above with the default cube on top of it. Rotate your view of the cube. The picture disappeared! But don't worry - the picture is still there. Press **NUM7** to realign the viewport and see the picture again.

Tip

Be sure to be in orthographic view **Num 5** or else you won't see the picture.

Now load the front view of the wolf into the **NUM3** (right side) viewport (YZ) as you did the top view. Repeat the procedure and load the side view of the wolf into the **NUM1** viewport (XZ). As a note, each picture is specific to the viewport it was loaded into. When you switch between axes views, the picture will not change. Try this out by changing the top left viewport to **NUM3** (right side) and **NUM1**. Return the viewport to **NUM7** (top) before continuing.

Noob note: the photo from the top goes into top view, the photo from the front goes into the right view, and the photo from the right goes into front view, if you have front in front, change it now!

The pictures are now loaded into the Blender viewports. If you look at the pictures on the grid, you may notice that the front view of the wolf isn't quite center. That is okay, Blender has a way to fix it.

Move your cursor over the **NUM1** (front) viewport and press **Shift + SpaceBar**, this expands the current viewport (NUM3/front view) to a fullscreen view, then open the Background Image window for the front view again. Notice that there are picture manipulation options available. One of these includes picture offsets. Click on right side of the 'X Offset: 0.00' to increase the offset to 0.20. The picture will be shifted over slightly so now the wolf is more centered. Press **Shift + SpaceBar** again to return to the four viewport view you created earlier.

Noob Note: If you made, and aligned your own photos, you will need to use a different value than '0.20' to center the head in the cube in front view.

The setup work is now done! Let's start on actually making the wolf model.

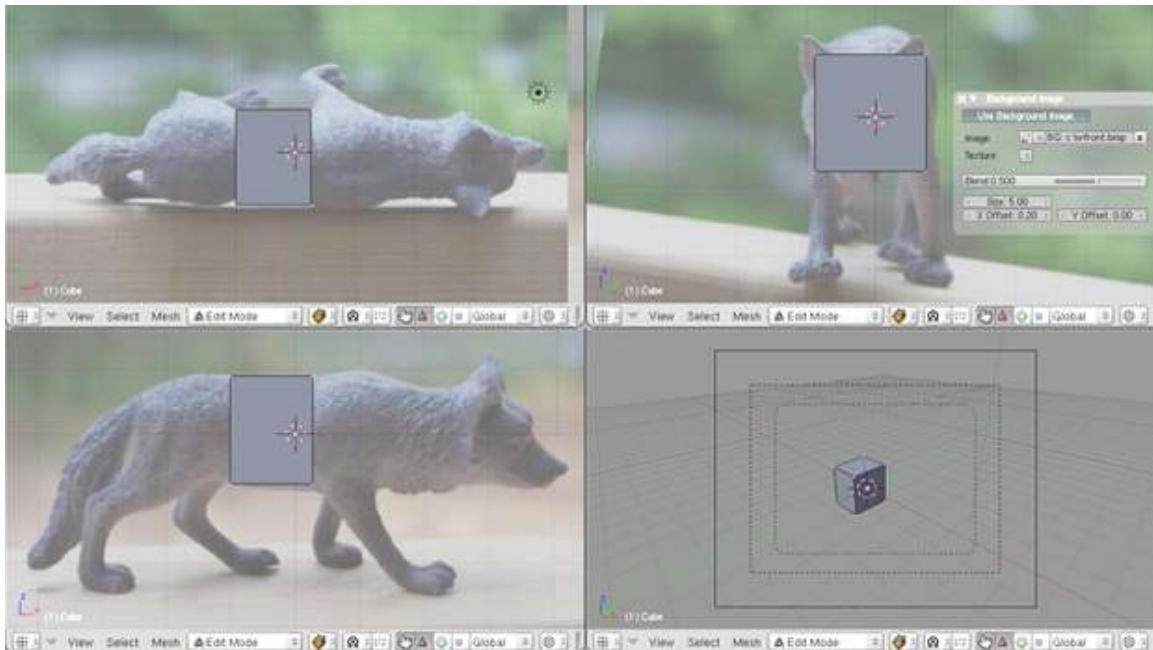


Figure 542

65.1.3. Create a Rough Model

This is a brute force model creation using techniques discussed previously in this book. This section is meant to help you explore and become more comfortable with them. Do *not* try to follow the example to the tee. Your wolf and my wolf will probably not look the same since you may want to add more or have less detail.

Noob Note: If you are just getting started with blender, this step may likely take several hours to complete, since you must use your own creativity to position things in 3D space, using 2D views. Just like sculpting, drawing, or oil painting, it will be extremely frustrating at first, but once you get used to the way it works, "modeling" will be much easier.

The rough fit stage requires either some planning or on-the-spot decisions. Think about where the wolf will have parts of its body flex or require parts jutting out.

The first step is to create a blocky wolf. Start out with a column of blocks using the extrude face command (select face, **EKEY**). Don't worry about snapping the vertices to the grid since we are working with an organic figure.

user note: I used **ZKEY** (wireframe view), to see better how the rough model fits out

Figure 2.3.1 Body column formation

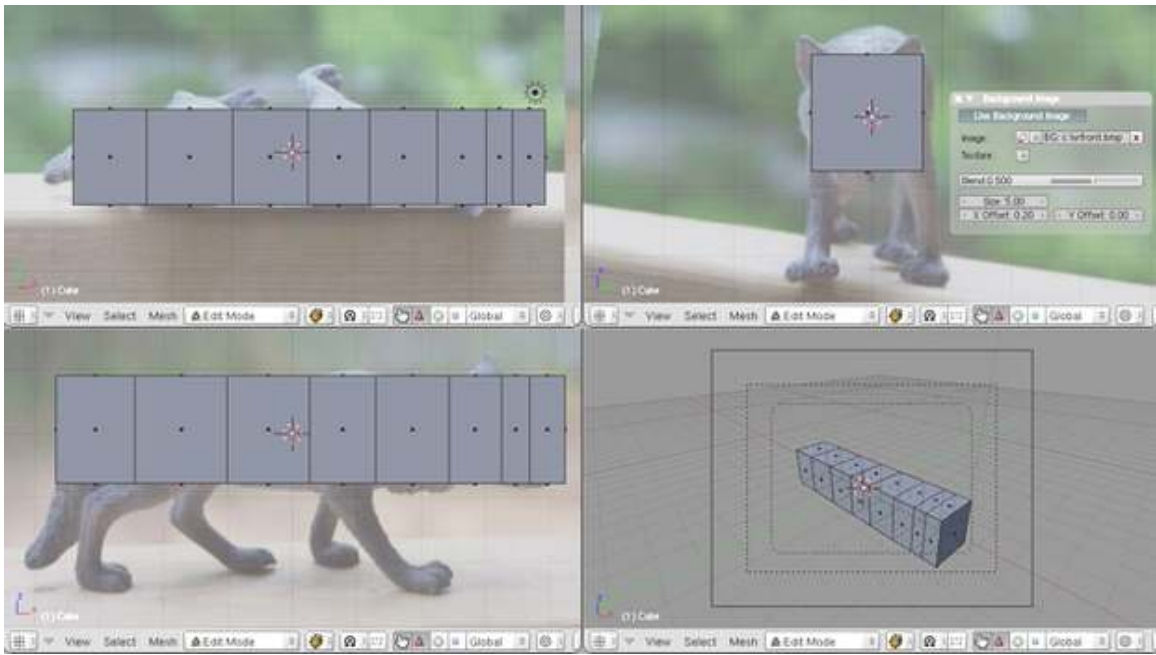


Figure 543

The next step is to split the ears and legs off of the body. Do this by subdividing the appropriate faces. Save often, and if you make a mistake, go ahead and use the undo option (**CTRL Z**KEY). Also, if you find yourself looking at redundant faces, combine them (**F**KEY).

Figure 2.3.2 Appendage formation

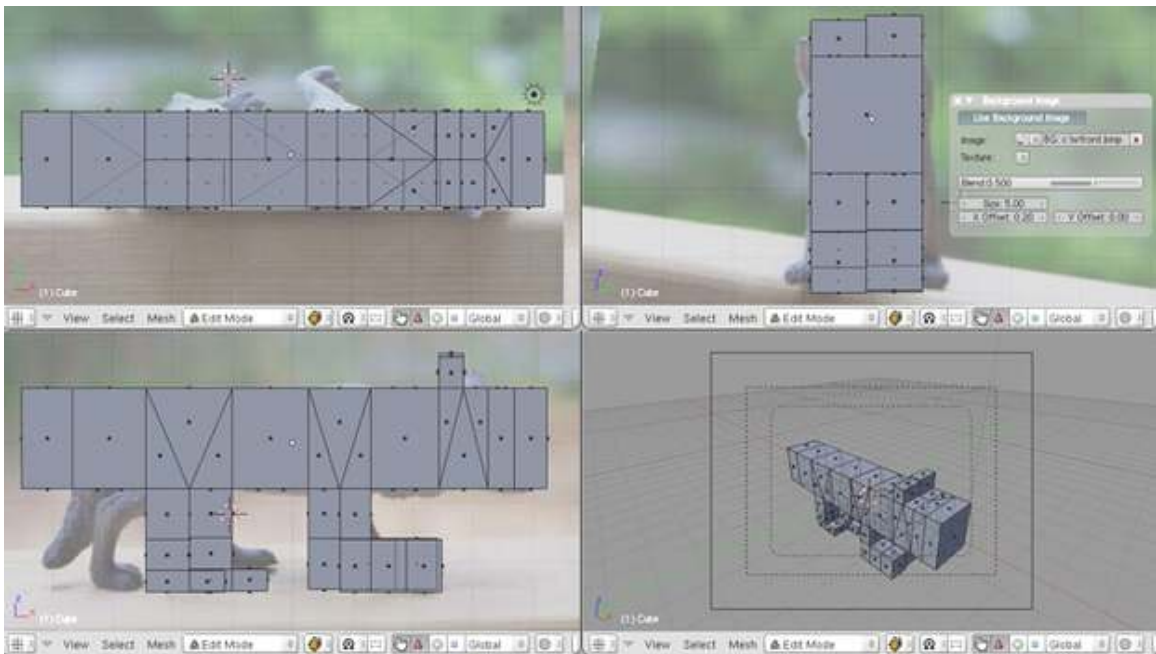


Figure 544

]

If you are having trouble with this, try mousing over the perspective window (the one you designated with NUM0) and using the MMB to rotate the view so that you are looking at the underside of the wolf. Click on the face underneath the wolf that is alongside his front legs (use the side view to check this). We are going to subdivide this face in order to grow legs off the new faces. To subdivide, press the WKEY and choose subdivide. You will see that the face has been divided into four. Take one of these faces and extrude it as many times as is necessary to make the right leg. Then do the same again for his left leg.

Noob Note: To avoid making the extruded faces share common vertices (and be connected to each other), either extrude the legs separately (as stated above) or select *Individual faces* in the Extrude pop-up menu. The same works for the ears.

Doing the ears is similar, except instead of working underneath you will start with the face on top of the wolf which is directly over the ears. Select this face and subdivide it once. Deselect everything using AKEY, then select one of these four faces and extrude it upwards once to make an ear. Do the same for the face alongside it to make the other ear.

Finally, extrude the tail end of the wolf one more time, so that your wolf has as many divisions as the picture above.

65.1.4. Refine the rough model

Let's start refining the model starting with the tail. Try putting your viewports in wireframe mode by pushing Z, it may make things much easier. Line up the vertices over the wolf in each viewport by lasso selecting multiple vertices (**CTRL LMB, Drag**). then move to the right location with grab (**GKEY**).

Figure 2.3.3 Working on the tail

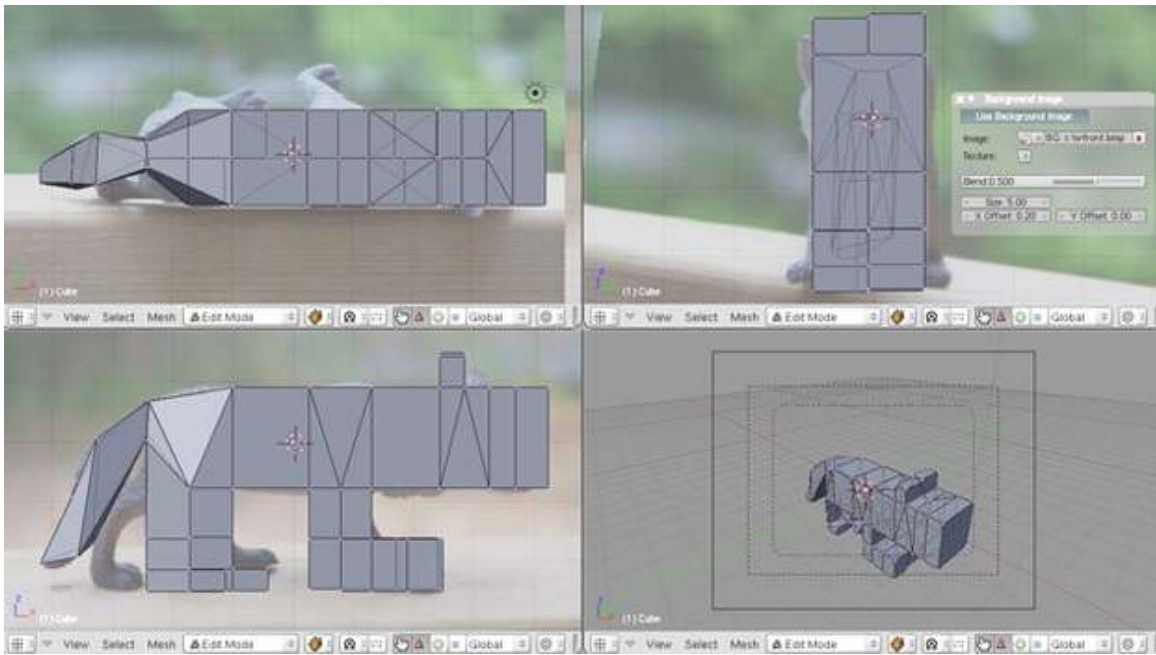


Figure 545

Continue onto the hind legs of the wolf. It is trickier to manipulate the legs so keep rotating a viewport to look at the model from multiple perspectives. Remember that we are working in three dimensions.

Figure 2.3.4 Working on the hind legs

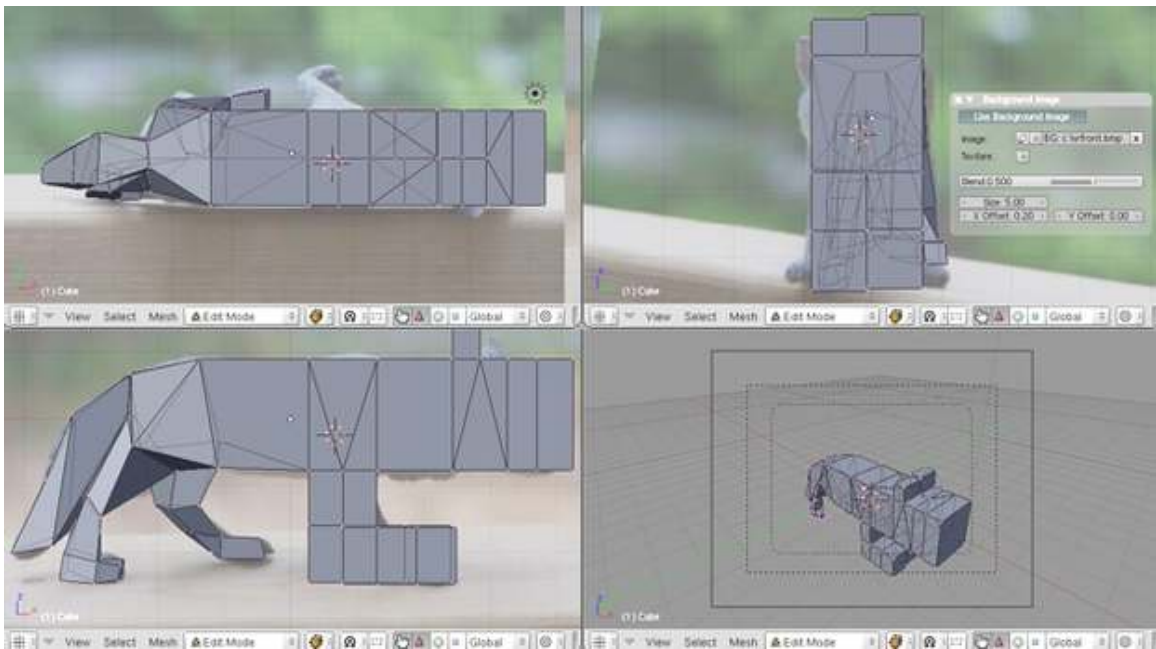


Figure 546

Continue working up along the wolf fitting the blocks to the pictures. If you have problems seeing the picture because the model is in the way, let's hide the model. In Edit Mode, select the entire model by **AKEY** or by pressing **LKEY** when you have the cursor over the model. Simply pressing **HKEY** will hide the selected items. To unhide the view, use **ALT+HKEY**. By hiding and unhiding the model, or parts of the model, you should be able to keep using the picture as a guide. *Note: It is much easier to just switch layers by, let's say, pressing **2KEY** (not the Num-Block one) to hide the entire model (thus getting a look at the picture) and **1KEY** to reveal the mesh again.*

Once you have the first pass done, you'll notice that the model just won't fit all three pictures correctly. This is due to parallax. The most obvious example is the side view. The four feet should be level, as they are all standing on a flat surface. Since they are not, we'll just ignore some of the aspects of each picture and continue with the model. (This is a helpful example to show what you need to consider when taking your own pictures.)

Figure 2.3.5 Completed rough fit

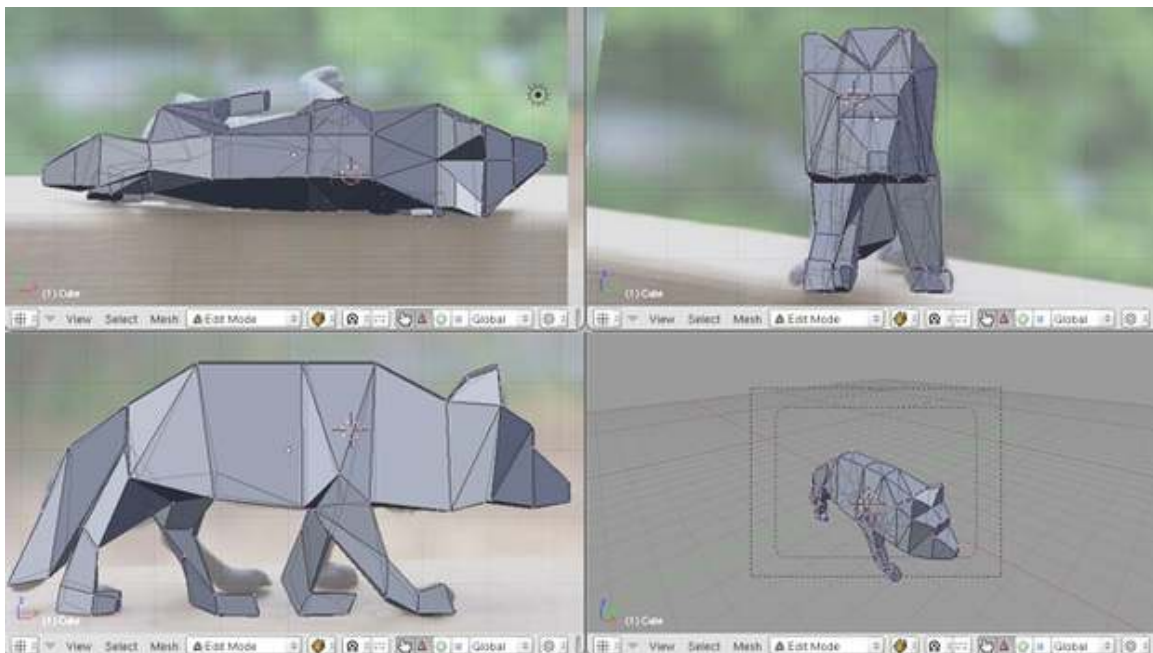


Figure 547

65.1.5. Subsurf the model

Now that the rough fit is done, let's smooth out the wolf. Add a Subsurf modifier and set the Levels to 2. The wolf will now be smoothed, but we want to add some of the hard lines back into the model. This may be accomplished with creased edges.

First, turn on the view creased edges by toggling the *Draw Creases* button in the Mesh Tools 1 window. Enter Select Edges or Select Faces mode (**CTRL TAB**). Highlight the edge or face you want to crease and press **SHIFT + EKEY**. Use the mouse and pull away from the center until the Crease value is close to what you want. A value of +1.000 will give you the sharpest look and is

useful for places such as the bottoms of the paws. When an edge has been creased, the edge will be highlighted in yellow (positive crease) or black (negative crease). These highlights are shown due to the 'Draw Creases' button being turned on.

In this example, I creased edges along the paws, tail, ears, and nose to give them some sharpness.

Figure 2.4.1 Creased edges

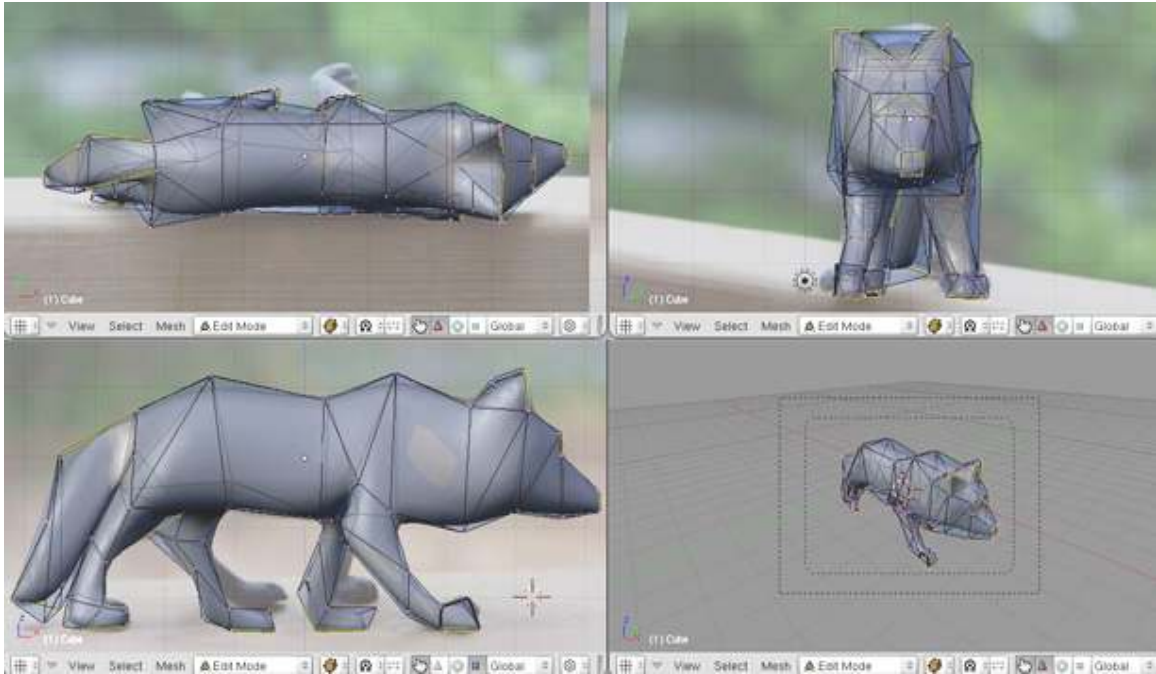


Figure 548

The last step is to refit the model to the pictures. You may have noticed that when the model was smoothed, the result didn't quite fit to the pictures. Now is a great time to tweak the vertices to fit to the pictures or add to/modify the model.

N00b note: if you see a weird edge on the body after smoothing, check the face normals (F9 > Mesh Tools More > Draw Normals; and then W > Flip Normals on the culprit faces).

And here is my basic wolf based on three pictures!

Figure 2.4.2 Final toy wolf model



Figure 549

65.2. Method 2 (using mirroring)

In this alternative method we will do half a wolf and then mirror it to create a complete wolf. You should start out by creating a silhouette of the wolf body in the side view, and then in front view start to shape the wolf head, tail and legs...

Noob Note: If you are attempting this method, it is best to avoid using triangles in rounded areas of a Mesh, because they tend to cause artifacts in your model (artifacts are protruding edges or other things that don't look realistic). Using quads is a much better alternative. Triangles should only be used on flat areas of your mesh if they cannot be avoided.

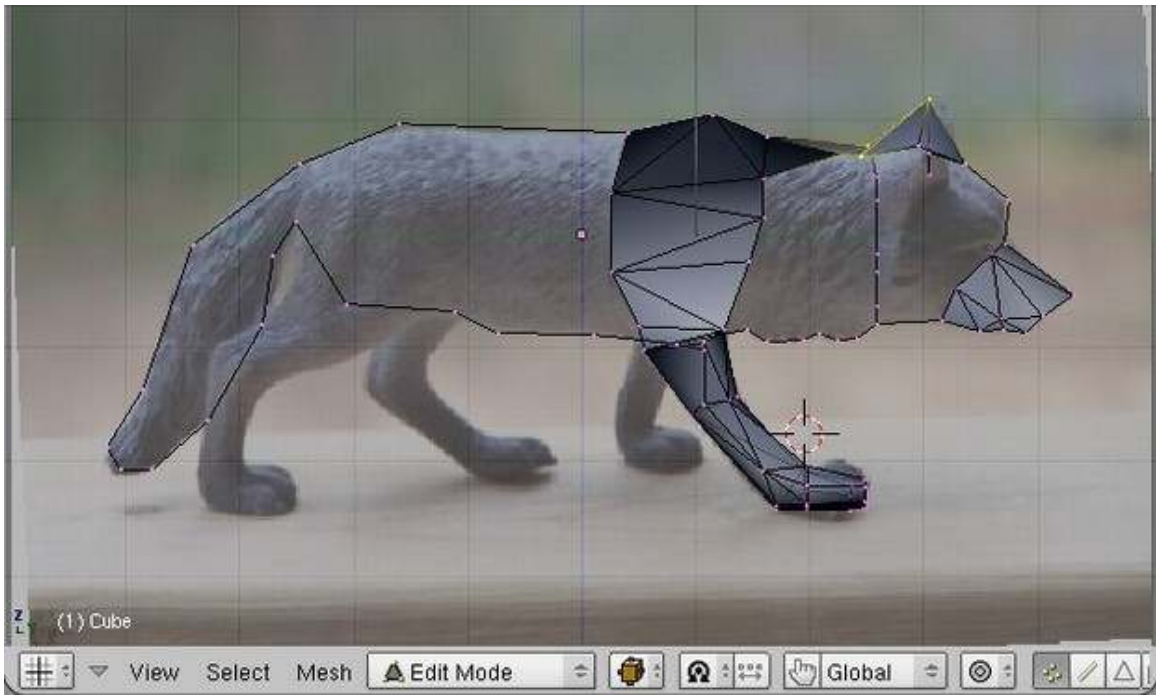


Figure 550

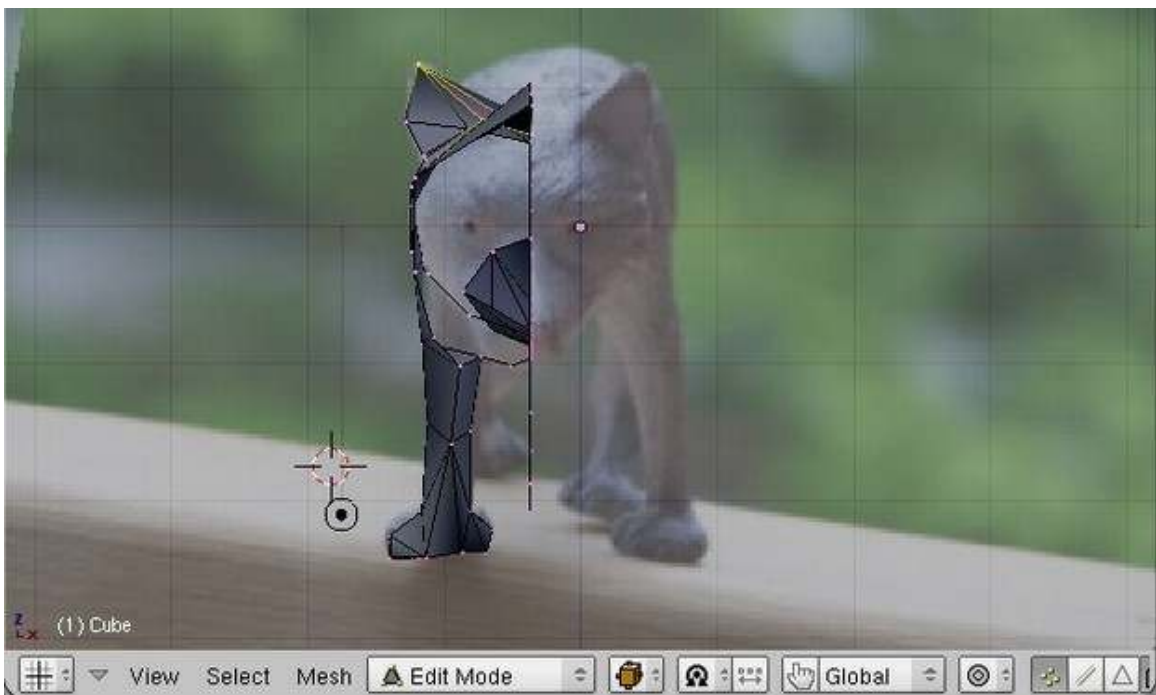


Figure 551

Here is what you should end up with

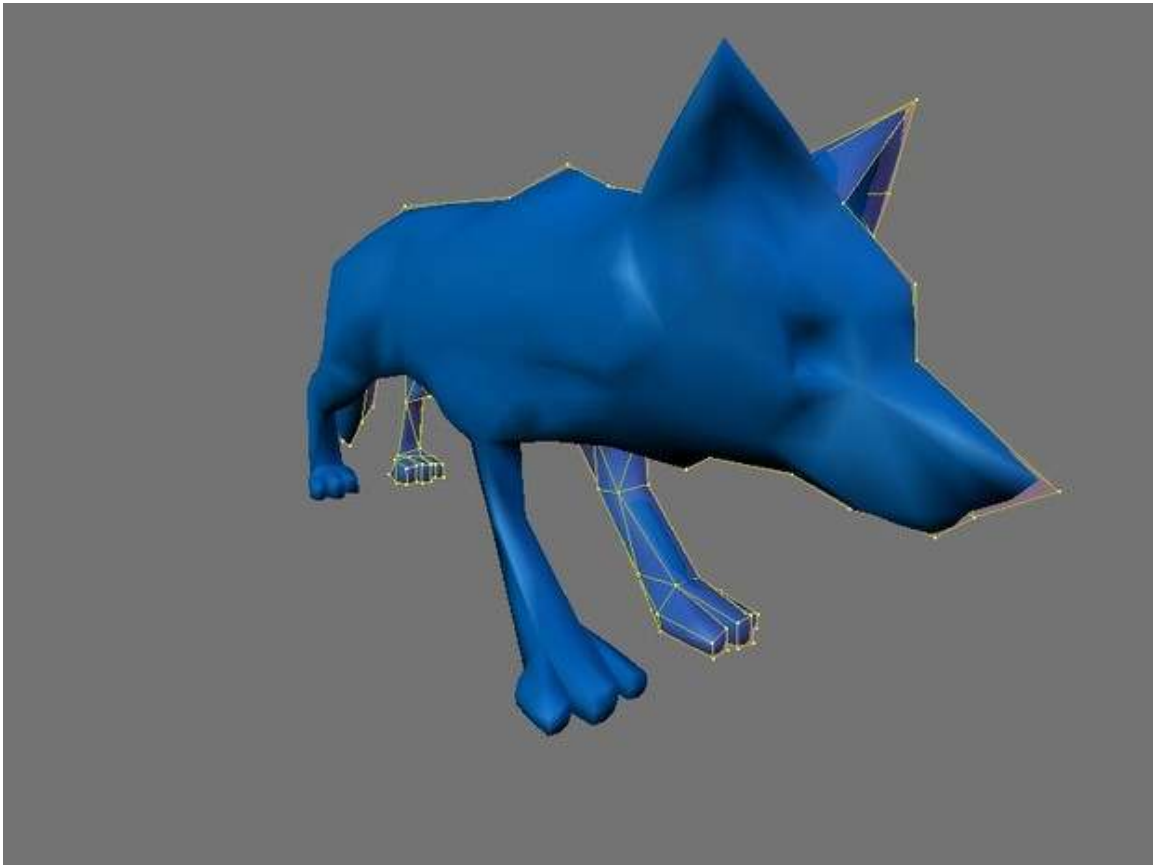


Figure 552

65.3. Using mirroring with method 1

You can also use mirroring with the first method (the one with cube extrusions).

This can speed up your work as you don't have to do the same changes on each side of the fox - just once.

To start the mirroring:

- Go to Editing (**F9**)
- In the *Modifiers* panel, add a *Mirror*
- Check only the *Y* button

When modeling:

- You may want to adjust the *Merge Limit* value (e.g. 0.14): this means vertices too close to each others (after the mirroring) will be merged, thus avoiding artifacts along the mirror's axis.
- Remove the faces along the mirror's axis, otherwise *Subsurf* will be confused by the faces *inside* the body of the mirrored fox. Keep all the faces for the legs though.
- To move the legs differently on each side: when you're done modeling the rest of the fox, apply the *Mirror* modifier, and then work on each leg independently.

65.4. Detailed steps to align images using GIMP

From this point forward there are multiple ways to do exactly the same thing, however for simplicity's sake and so that I can be more detailed I will be using one method (the one which I use) and be using GIMP.

I found it best to size all the photos to a known width, with an easy to find center. (Mine happened to be 850x638 pixels, I don't recommend that but you can choose any size you want really, as long as all of them are the same size). Then drag the construction lines to form a crosshair in the middle of the photo. To do this, click on the top ruler, and drag down to the middle (Exact middle) of the photo, then click on the side ruler and drag across to the middle (Again exact middle) of the photo.

If you are having trouble finding the exact middle of the photo, move the cursor to the very bottom left of your photo and the height of your photo will be listed at the bottom left of the GIMP interface. The numbers are listed in an (x,y) format so you want the first number to say 0 and the second to be the largest you can make it by dragging your cursor. The second number is the height, and half of that is the middle of your photo. You can do the same with the top ruler to find the vertical middle of your photo. Only this time the co-ordinates at the bottom left of the GIMP interface should list the second number (y) as 0, and the first number should be as large as you can make it by moving your cursor (to the upper right of the photo). Once you have your width again half of that will be the middle of your photo.

Then using construction lines put one at the top of your object, and the bottom of your object. Find the "height" of your object by the distance between them. Remove the construction lines from the top and the bottom, and place a new construction line above the horizontal center line by the half of the "height"(of your object). Now place a construction line on both sides of your object and find the "width" (distance between the new vertical lines), then remove those construction lines and place a new construction line vertically half of the "width"(of your object) to the right of the vertical center line. Now cut the object out, and drag it so that the point you used as the "top" is on the horizontal construction line that is above the middle. Then Drag the photo left or right until the right edge of the object is on the vertical construction line you put in right of the middle construction line.

Now the center of your object is at the center of your photo. This is a very important thing because when blender loads in the picture you will need this so that all of your pictures match up with each other 3d. You should repeat these steps with all 3 photos. I also don't recommend doing it in GIMP's "layered mode" as that caused more pandemonium for me. I recommend opening each photo in a new window .

Taking your pictures is the most important part, because if the pictures are not all in the same scale (object size to photo size) then your photos will not line up and you won't be able to place a dot on the same location from front view, side view, and top view.

As a recommendation I would recommend making your first model from a Lego man. That is what I did and it is very simplistic easy practice. To take my photos I took about 10 minutes to construct a photo platform for my object. It consisted of a cardboard box with two sides cut out. I covered the inside area with computer paper. I then used a 2"x4" and a ruler to make sure that the box stayed the same distance from the camera for all shots, as well as marking where the Lego man's feet were positioned inside the box with a pencil. This will provide good pictures, providing you keep the camera at the same distance and zoom for all three photos.

66. 2D Image (logo) to a 3D Model

66.1. Overview



Figure 553

We will be using this graphic as a template for a 3d logo, tracing it, then discarding the 2d image.
(Click thru to get the larger sized version, for you to follow along with)

66.2. Load the background image into blender

66.2.1. Basics of the background image panel

If you haven't already done so, open blender and select one of the orthogonal view angles by pressing **NUM7** (top), **NUM3** (side), or **NUM1**. (front). At the bottom of the 3D viewport on the left, there are some menus, click **View** -> **Background Image**

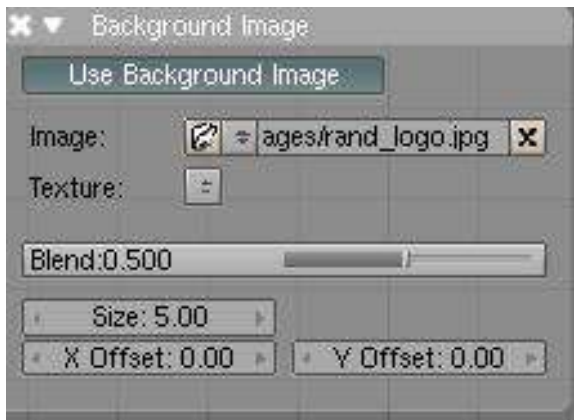


Figure 554

A small window will appear containing just one button marked **use background image**; click this button. A few more buttons will appear. One of them says **image** and has a small button with a picture of a folder on it; click this button. You are now presented with a file selection screen. Using the navigation techniques from the previous tutorials, find your 2D jpeg image on your computer, click the file in the list once then click the **Select Image** button at the top right of the screen.

Noob note: The image is only displayed in orthogonal view. If perspective view is enabled, toggle to orthogonal view by pressing **NUM5**. (*The image will not be rendered as it is not part of your scene.*)

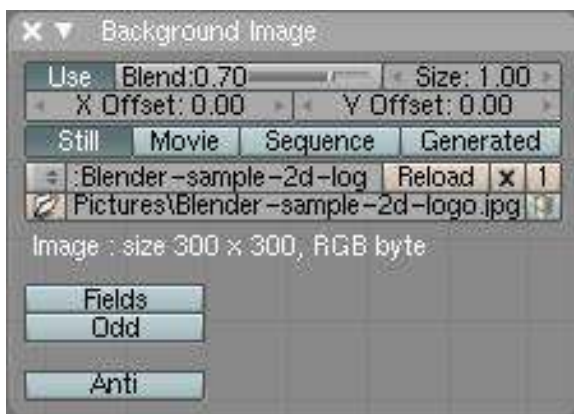


Figure 555

Once a background is selected you'll have a dialog like this one. (Note: This tutorial was originally generated from Blender v2.37. v2.43 has been added - older versions may differ.) The background dialog buttons are described below:

66.2.2. Details of the background image panel

The



Figure 556 Use Background Image Button v2.37

is a toggle button that turns display of the image on or off. Turning the button off will not clear the settings; it just hides the image. When you turn the button on again, your previous settings are back. Try it - click the button a few times. In v2.43 the equivalent button is the



Figure 557 Use Background Image Button v2.43

button.

Image selection is controlled on the row labeled



Figure 558 Blender_Image Select v2.37

. There are 2 buttons, a text box, and a final button. The first button

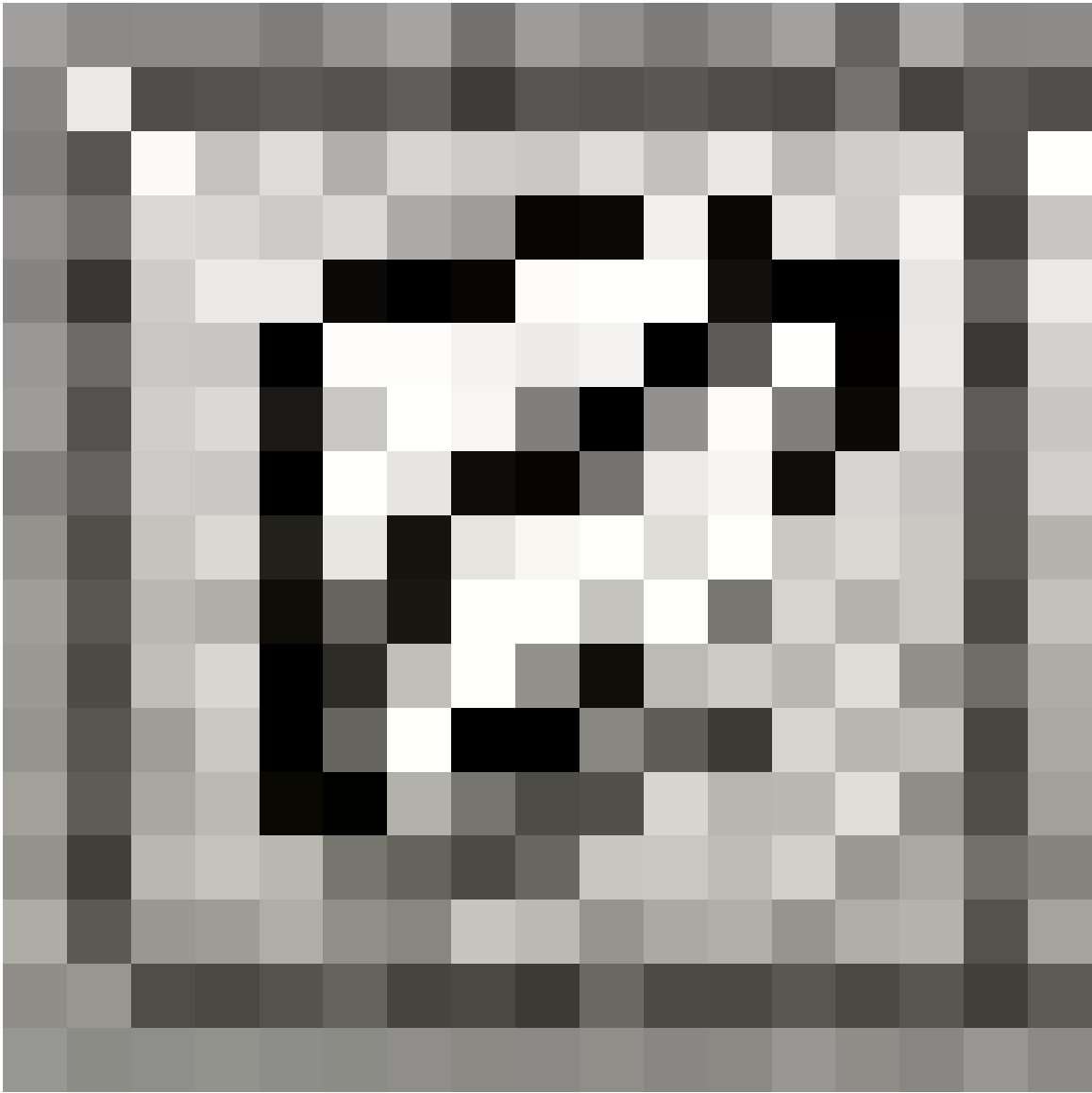


Figure 559 Browsing Image Select v2.37

is used for browsing for an image. The 2nd button

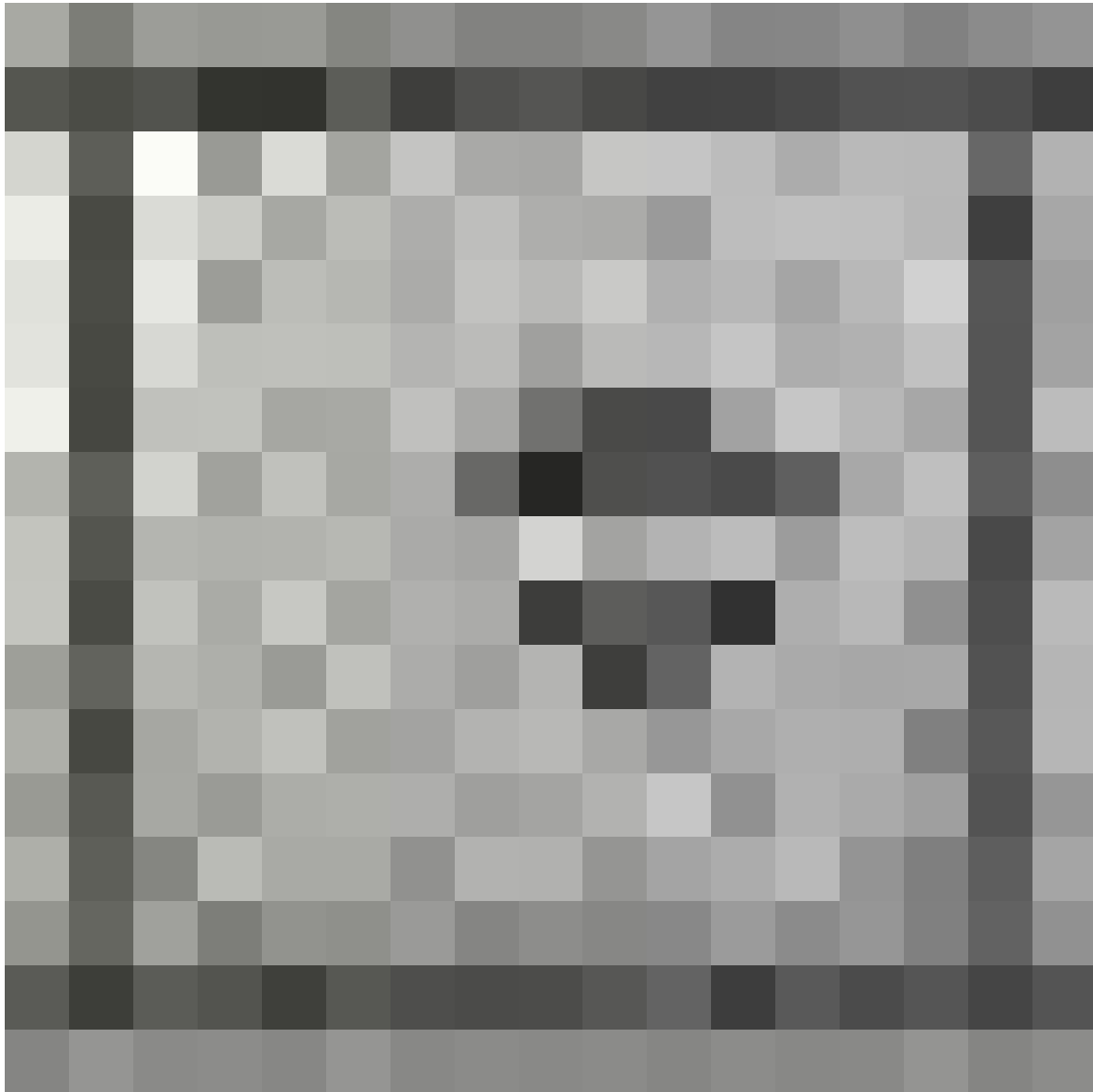


Figure 560 Image Select History v2.27

is for selecting an image from a history list. (This will be empty for the first time. Selecting it now will display the image you currently have selected.) The text box



Figure 561 Image Select Textbox v2.37

allows typing in the file directly. The

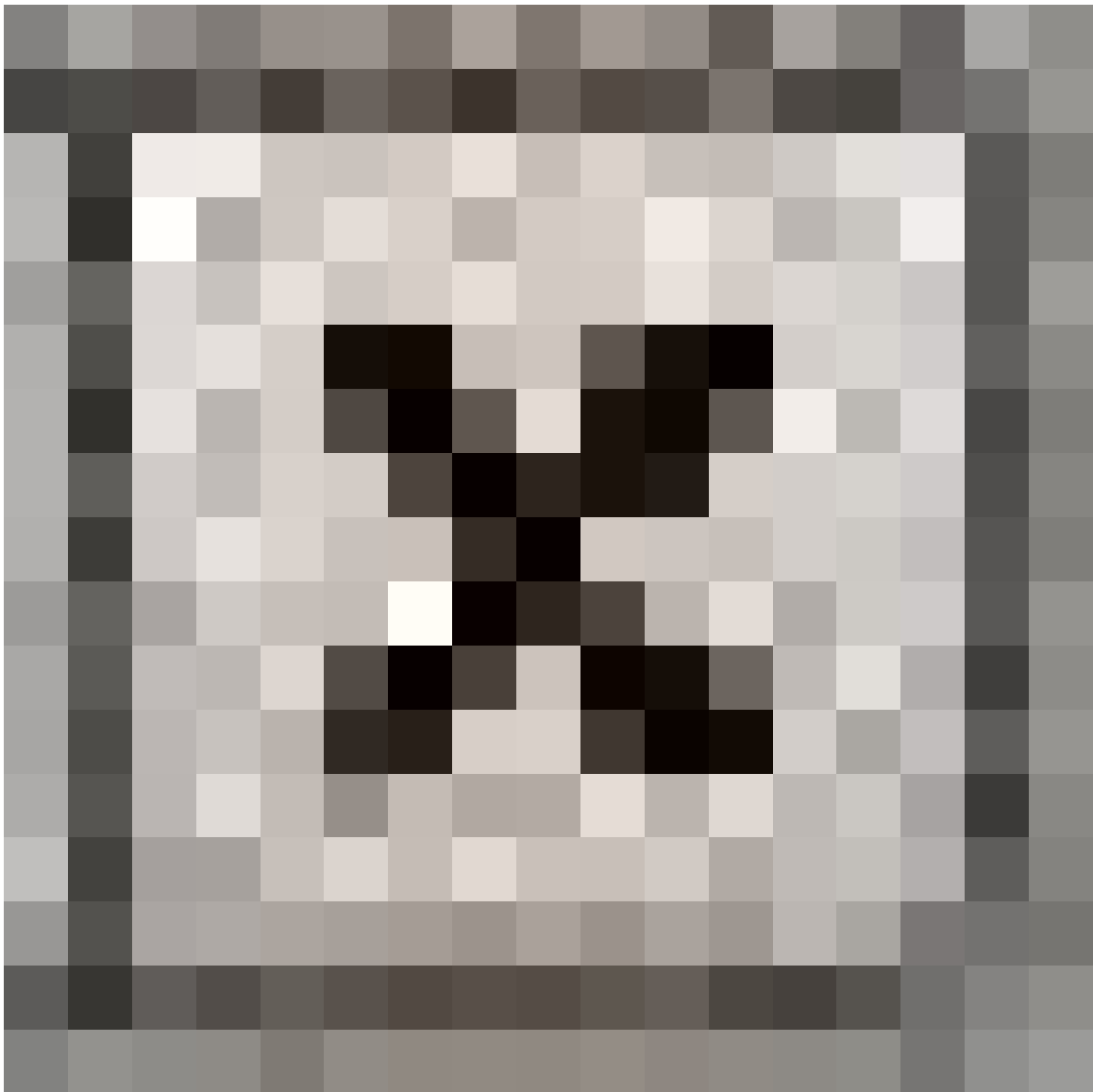


Figure 562 Image Remove

button removes the current background image. Version 2.43 is the same with the addition of the



Figure 563 Image Refresh

button that refreshes the image or movie, and the

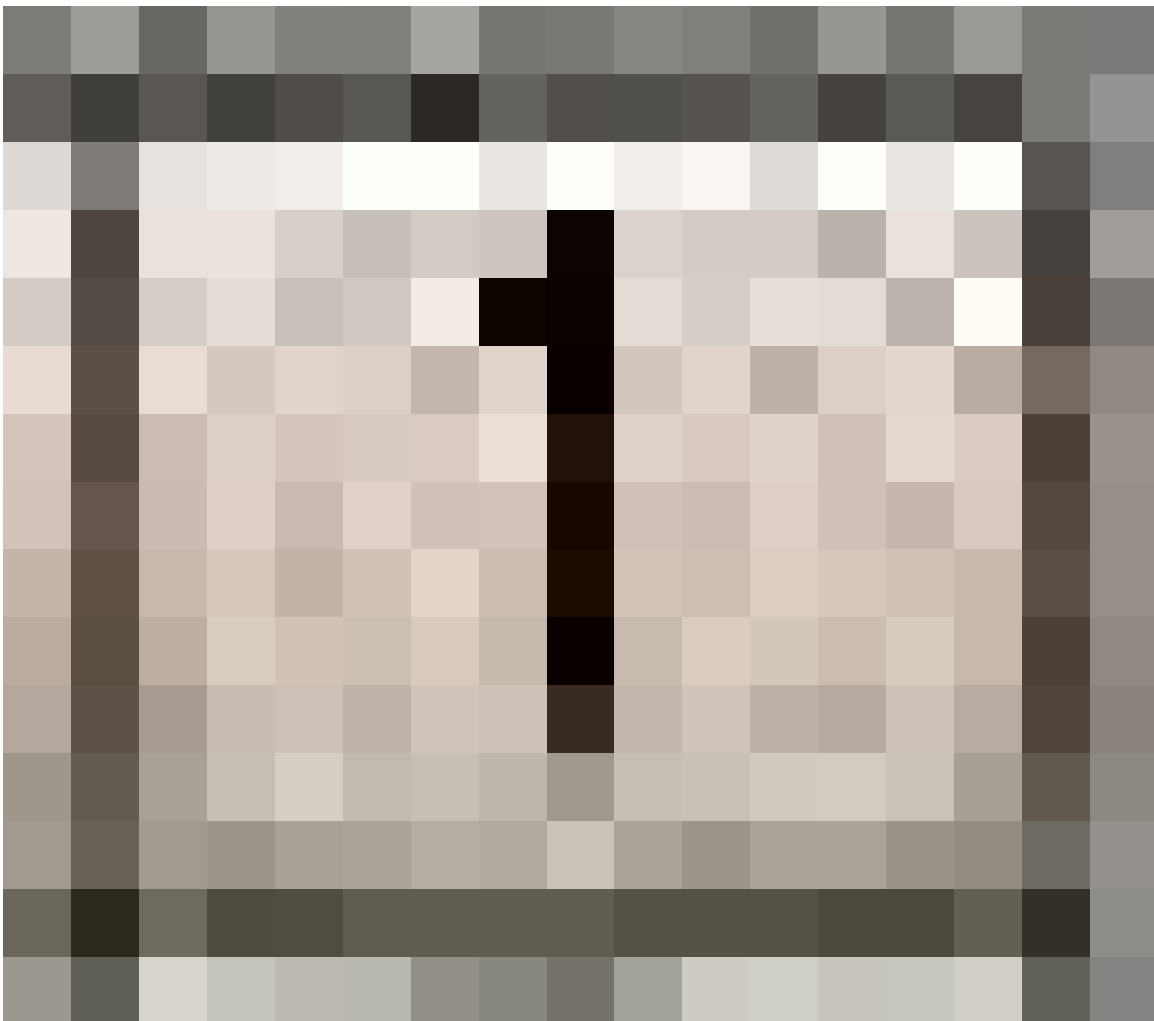


Figure 564 Image Users Count

button which shows the number of users of the image block.

The third row is called Texture and will not be used for this tutorial.

The fourth line, labeled **blend** controls the transparency of the background image with a slider. A setting of 0 is completely solid and 1 is completely transparent. You can adjust it by clicking left or right of the knob for gradual changes, clicking and dragging on the slider for rough settings or clicking directly on the **blend** text for numeric entry.

The use of the blend function will become obvious once we start tracing our logo. For now, play around with it, see how it changes the image, and put it back to the 0.500 default.

The fifth line, **size**, controls the size of the image. This size setting is independent of the zoom for the 3D view window. To see how the size works move the default cube off to the side so that you can see both the cube, the background dialog and the background image. Now watch both the cube and image as you change the size. Notice how the image changes size but the cube doesn't? Now press **NUM+** and **NUM-** to change the view's zoom. Now both the cube and image change size.

The final row controls the X and Y offset for the image. These controls move the image up and down (Y) or left and right (X). These settings can be useful if you need to reposition the image from the default position. Like the size, these offset values are independent of the view. As you change the offset values the cube you added earlier won't move. Now scroll the view using by clicking and dragging the **SHIFT MMB** and notice how the cube and image move together?

Once you start tracing the image you won't be using the size or offset setting.

66.3. Delete the cube

Delete the cube (select it, press **XKEY** and select **All** from the **Erase** menu), and set the size so that the entire image is viewable. Then set both the X and Y offsets to 0.

Set the view to top view (**NUM7**) for the rest of the tutorial!

Finally minimize the Background Image dialog. You'll only need it to adjust the blend setting until you finish tracing.

66.4. Introducing the Bezier Curve

The Bézier Curve¹ allows drawing graceful, complex curves and only requires a few points. Specifically, it only requires 4 points for a curve: two end points and two control points.

For the moment set the blend to 1 on the Background Image dialog. With the center of the 3D view still selected, press **SPACE -> Add -> Curve -> Bézier Curve**. Alternatively you can use the **Add** menu at the top of the screen or press **SHIFT -> AKEY** to jump directly to the add menu. Be sure to be in **EDIT MODE** not **OBJECT MODE**.

¹ <http://en.wikipedia.org/wiki/B%26E9zier%20curve>

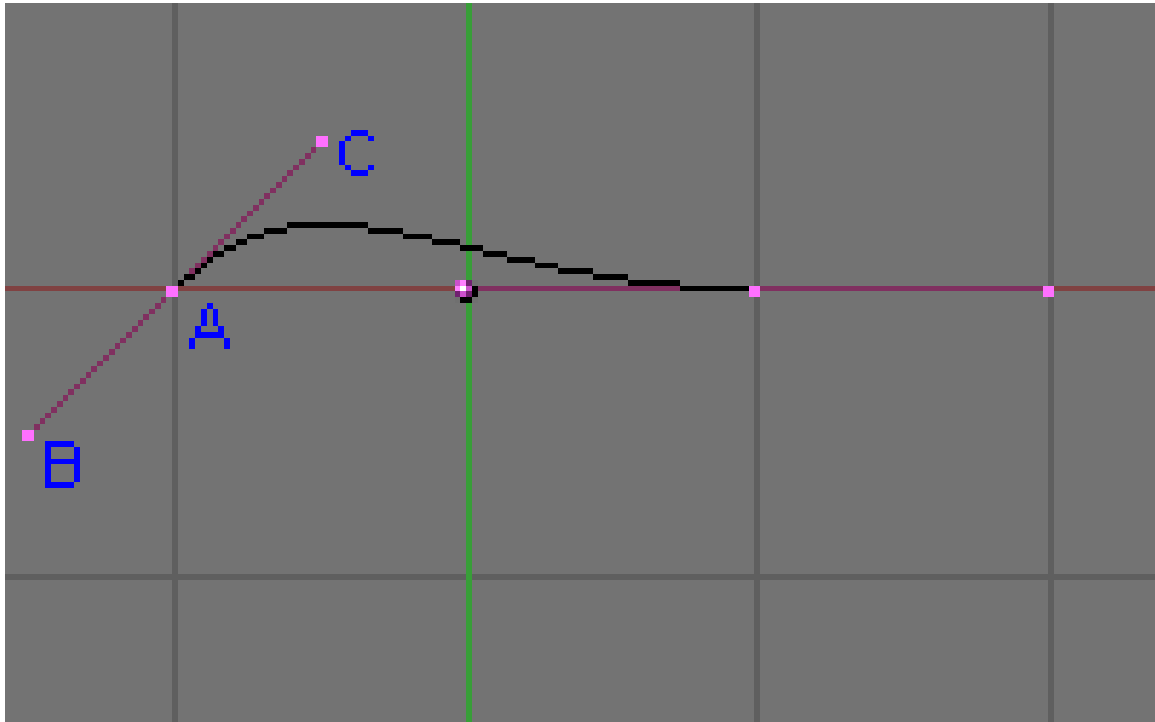


Figure 565 The Curve you added.

Unlike the traditional Bézier Curve each Bézier vertex has 3 points. I've labeled the 3 points for the left vector: A, B and C.

Just like any other vertex you select the control points and the end points with **RMB** and move them with standard commands like **GKEY**, **SKEY**, **EKEY** or **RKEY**. if you wish to rotate, scale, or move the whole curve, then select all the vertices on the curve using one one of the various selection methods, and then use the **GKEY**, **SKEY**, or **RKEY**

Point A is an end point. The curve will always go through this point. **Points B and C** are control points. These points influence the path of the curve as it leaves **Point A**. Because the path stops at **A**, **Point B** has no real effect on the path. Instead **B** is currently locked with **C**. (If you move either **B** or **C**, the other will move.) We will *fix* **Point B** to move independently a little later.

The control points have 2 effects on the path exiting the end point: direction and distance (these are termed slope and magnitude in math circles) from **Point A**. The direction will provide the direction that the path will follow when it leaves **A** and the distance will determine how long the path follows that direction before it starts making its way to the curve's next point.

```
<!-- these images never seems to have been uploaded
```



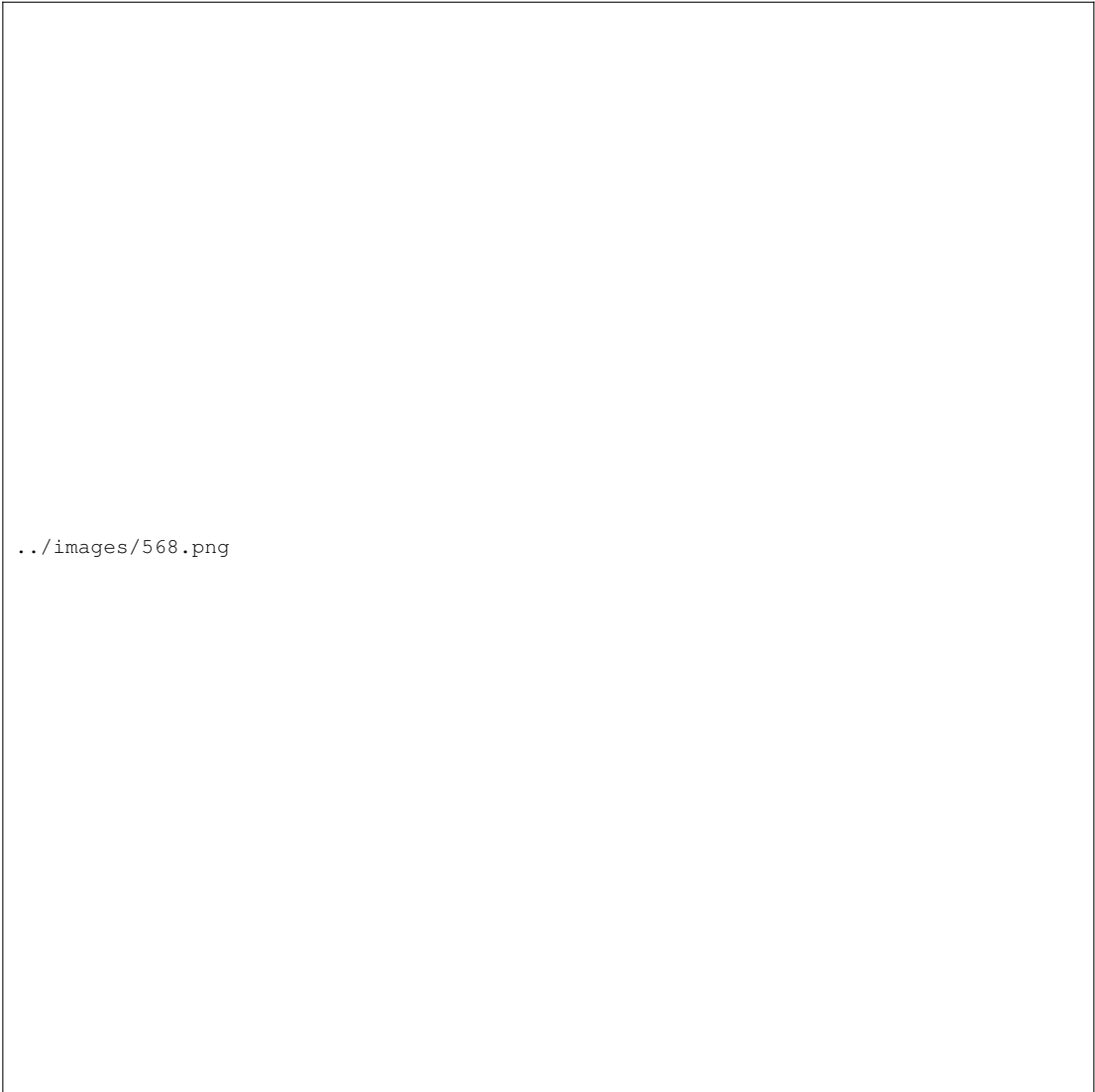
../images/566.png

Figure 566 N.B. To easily make sharp turns; take *B* point and move it really close to *A*.



../images/567.png

Figure 567 Take *C* point and move it in the exact direction you want to turn



../images/568.png

Figure 568 (with *C* selected) Press EKEY to extend the Bezier, and drag it out; you will find it goes where you pointed *C*! --Jawboot^a 01:30, 30 April 2006 (UTC)

^a <http://en.wikibooks.org/wiki/User%3AJawboot>

-->

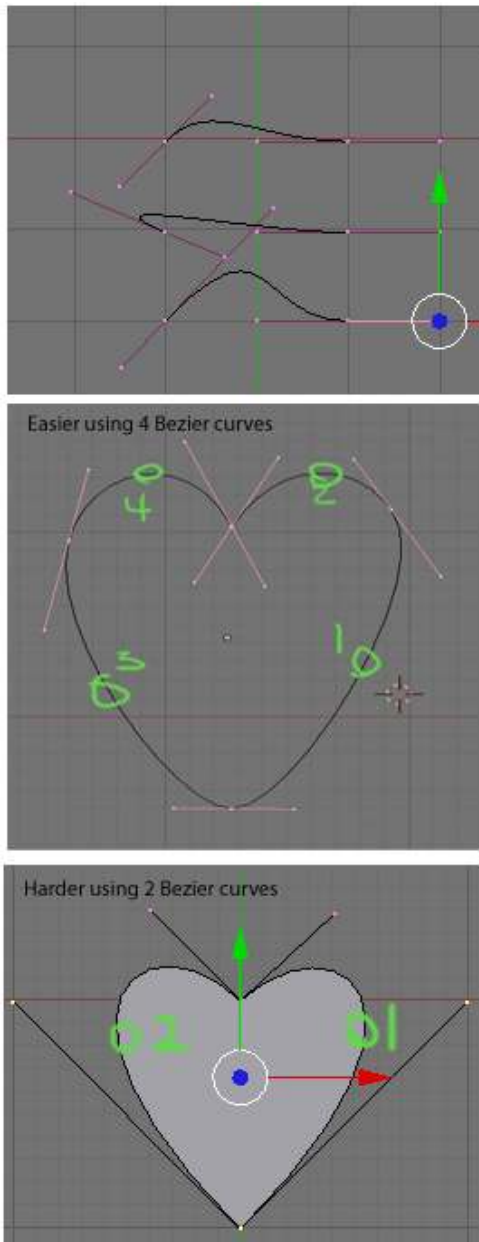


Figure 569

The example to the right shows how the control points influence the path of the curve. In the top picture, we see three curves. The top curve is the default curve. In the next curve down, **C** has been moved to give a drastically different direction. Notice how the path leaving **A** moves away from the other end point. The third curve, the distance was changed dramatically. Watch the path move much higher than the other two curves.

In the bottom example, I've built a heart shape using just the points shown. *Dragging the bottom end point down will make the shape closer to a leaf.* You'll be able to do the same at the end of this tutorial. Go ahead move around the points for the curve and see how they all interact. Get a good feel working with the curve and when you're ready we'll move on to tracing.

Now that you know how to work with a bezier curve set blend back to 0.5 on the Background Image dialog so we can start tracing.

66.5. Modeling the lighting bolt

66.5.1. Rough Tracing

If you don't already have a curve add one now. It will help to move the curve to the center of the yellow lightning bolt.



Figure 570 Curve Tools for Blender 3D version 2.37

The first step in tracing is to click the Polygon convert button on the curve tools panel. You'll find this in button on the **Buttons Window**. You may need to select the **Edit Panel**. Press **F9** if this panel isn't visible. In Blender 2.59, the 3D View panel's Tool Shelf (**T**) houses the Curve Tools, including Set Spline Type, from which you should select Poly. You must be in Edit Mode to reach the Curve Tools.

Noob Note: The 'Curve Tool' panel will not appear if you don't have a bezier curve already placed. I learned this the hard way.

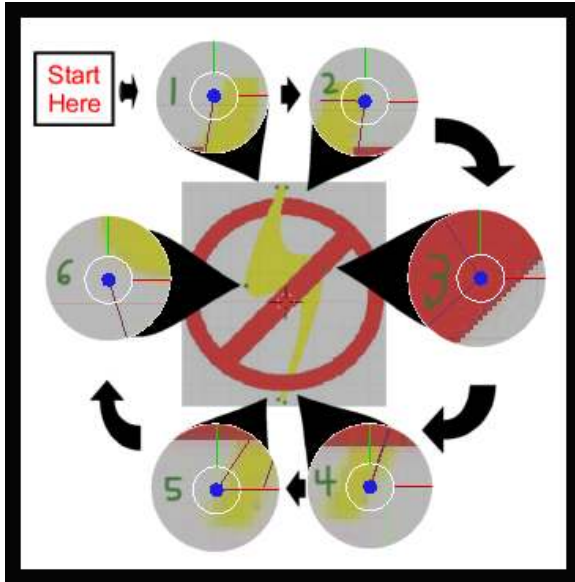


Figure 571

Next, move the vertices of the curve to the points shown in the image to the left. This is called Rough Tracing because you don't need to exactly trace the image.

You only need to approximate the image. Moving the vertices should be done using the instructions from the Creating a Simple Hat² tutorial.

Note: Selecting the best place to put a vertex is a bit of an art that you'll acquire as you work with curves. For now follow the arrows along the cutouts and place each of the vertices as shown.

This tracing uses all the vertices of the polygon. Other cases, you'll need add or remove extra vertices by selecting the end point of your curve press **CTRL** and click **LMB**. At the place you clicked a new vertex will appear connected to your curve.

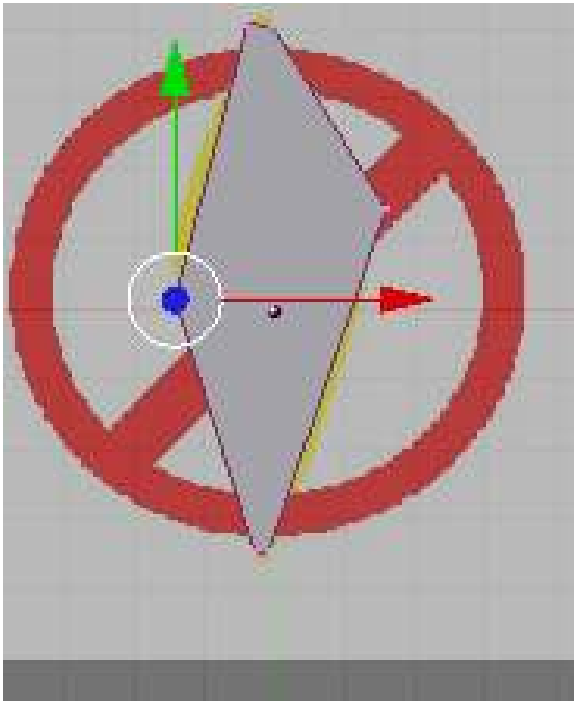


Figure 572

After moving the last vertex, we finish the rough tracing by pressing the **C**KEY to close the polygon. You should see an image similar to the one on the right. (If you only have an outline switch your view port shading to solid by pressing the **Z**KEY for now.)

Question: how do you close the polygon in blender 2.61? **Answer:** ALT + CKEY pressing c key alone brings up circle select.

Note: In Blender 2.63 and certain prior versions, the curve needs to be changed from 3D to 2D shape in order for the polygon to show up, else you get an error. This can be changed by going into Object Data and, under the Shape heading, switching the curve from 3D to 2D.

Notice how the polygon doesn't cover all of the yellow of the bolt and how in some places the polygon fails to conform to the shape of the bolt. This is expected and should not be a cause for concern. We correct this in the next section.

Once you've finished several logos you should begin to get a feel for the required placement of vertices. Until then, here are some general guidelines to keep in mind:

- A gradual curve may only require a single vertex.
- Tight curves will likely require two closely placed vertices.
- Curves may not require a vertex at all - you can define some curves using the control points of the adjacent vertices. We did this for both of the inside curves of the bolt above.
- Corners require a single vertex placed where the curve bends. A square, for instance, requires four vertices - one at each corner - to be modeled properly.
- The end point of a curve will always be on the curve. So should all of the vertices you place.

We are now ready to move onto the next step modeling the logo.

66.5.2. Polishing the Tracing

Press **ZKEY** to return to wire frame mode, then Press the **Bézier** convert button to convert the polygon back to a curve.

This will convert your polygon back into a curve. Nothing obvious will happen. If you look close, you should notice the number of points on the curve tripled. When you converted the curve back to a Bézier curve, Blender changed all of the polygon vertices to Bézier vertices. While the polygon vertex is a single point the Bézier vertex is made of an end point and 2 control points. So the extra points are the control points of the Bézier vertices. These control points are placed along the curve to produce the same shape as the converted polygon.

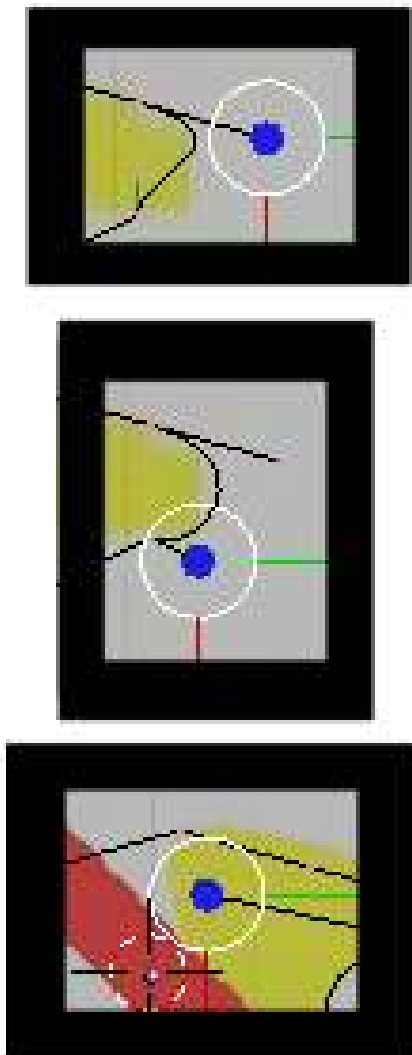


Figure 573

Our job is to move the control and end points so that the curve follows the edge of the bolt. The trick is to move the 2 control points between adjacent end points to bend the curve to the edge of bolt. First, move the right control point of the top-left vertex. This should pull the curve from its end point

to more closely match the line of our bolt. After placing this point, we move to the next control point following a clockwise direction around the bolt. Use the **RMB** to select the point you want to change and move it with **GKEY** to place it.

Noob Note: if you wind up with pink lines when you select the control handles, that means you hit **HKEY** by accident, and have put the control point in manual. to correct this hit **HKEY** again to put it back into "easy" or auto mode.

As you move the second point notice how the curve exiting the first end point is drawn away from the edge of the bolt being traced. We now have to adjust the first control point again to get that line back on track. This quickly turns into a balancing act adjusting each set of control points. The trick is to make smaller movements for each iteration of adjustments. Make a game of it and move the control points all along the bolt. Always move along the clockwise direction. This practice is not just for consistency, it keeps your place and ensures that moving a control point doesn't change a portion of the curve that you've already completed. In time you learn to move the first point only part of the way. Then moving the second brings the curve for the first into correct alignment.

If you have some trouble aligning the curve to the edge of the bolt, consider adding a new point. There are two (at least) ways to accomplish this:

- Select 2 points that surround the problem spot where you want a new vertex and click the Subdivide button on the Curve Tools 1 tool panel.
- If near an end point, Select it, press the **CKEY** to open the curve, then **Control+LMB** click to add a new point beyond the end of the selected final vertex. Press the **CKEY** to reclose the curve.

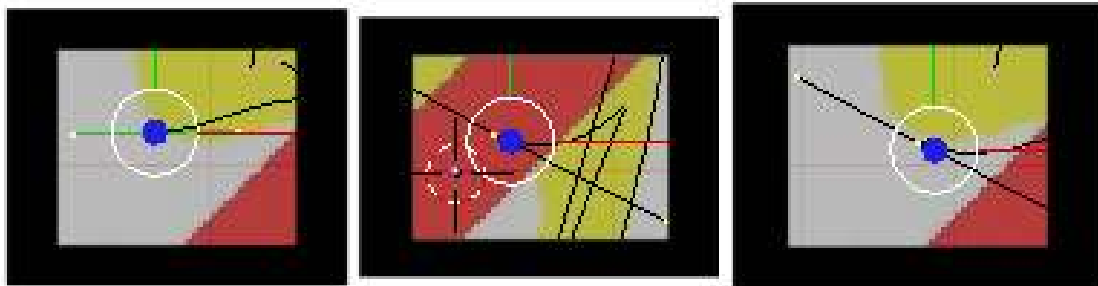


Figure 574 Adding a new end point

The new end point should be positioned and then you have to adjust the curve on both sides of the end point you move. Any time you move an end point be sure that the curve going into both adjacent (clockwise and counter-clockwise) end points still aligns with the edge of the bolt.

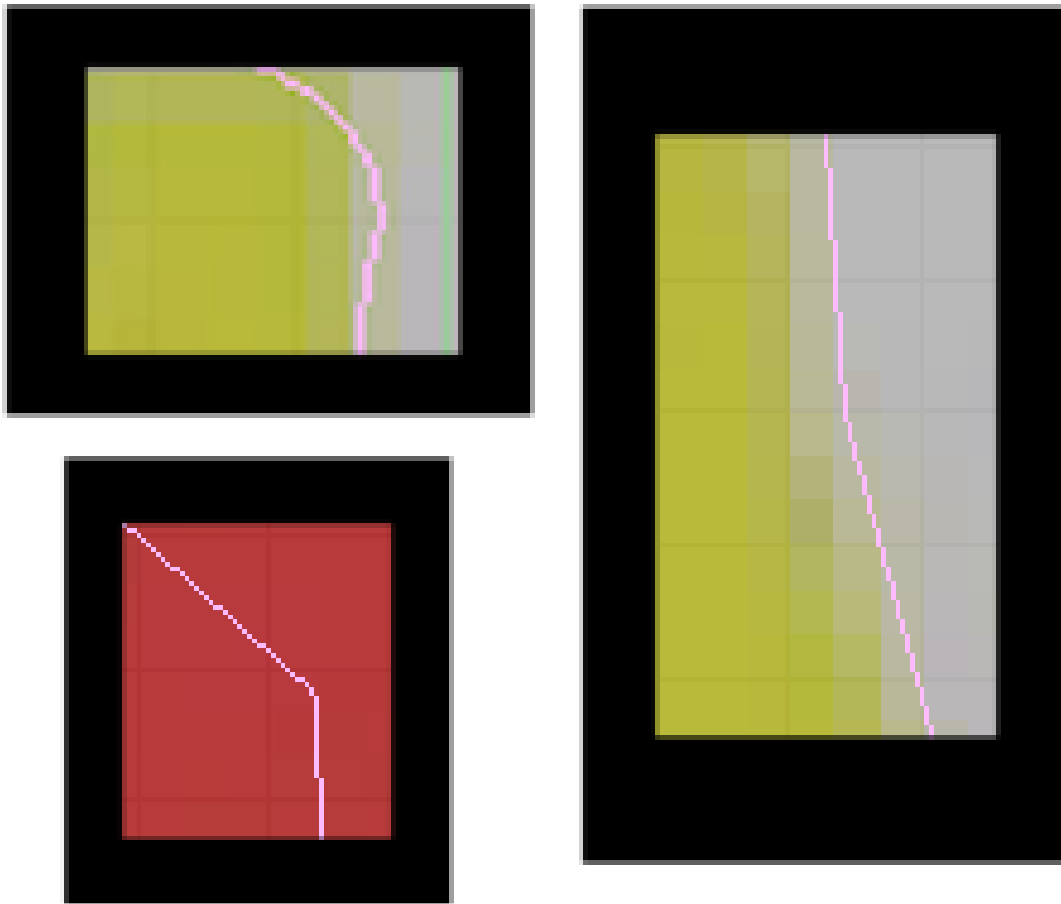


Figure 575 Finding trace defects

Once you've made the complete circuit around the bolt, you're ready for the final polishing of the edge of the curve. Press the **TAB** to switch to the object mode. This makes the polishing easier as Blender hides the points and lines for editing the curve. Now zoom in on the bolt's edge using the **NUM+** or **Control+LMB** drag. Use **Shift+MMB** drag the screen so that you closely observe the entire edge of the bolt while zoomed in closely. Look for places where the curve pulls away from the edge.

Also look for sharp bends at each of the end points that should be smooth. You can see several defects that I found in my project after tracing the bolt. Switch back into edit mode to fix the curve and then go back into object mode to look for more defects.

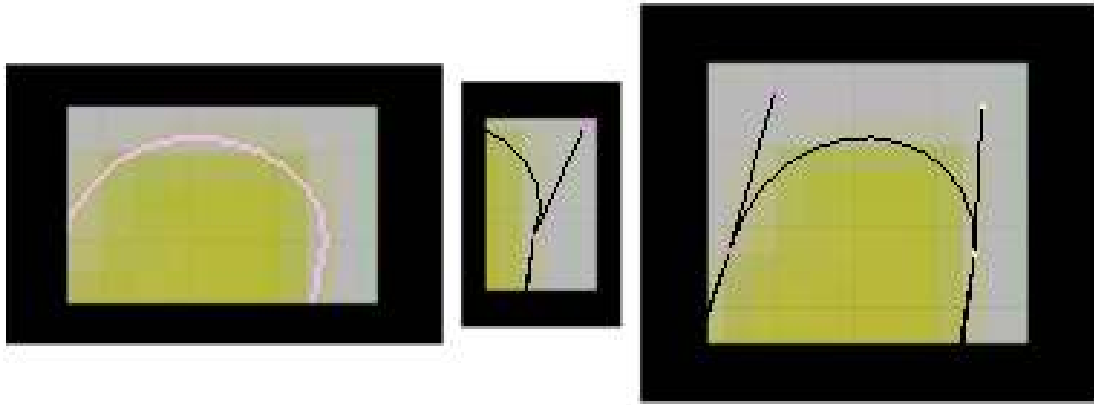


Figure 576 Correcting poorly shaped curve

Sharp end points are adjusted by decreasing the angle between end point and the control points. Many times that is impossible to do, without messing up what you're trying to draw, so the other way is to add more curves by subdividing (**WKEY** -> subdivide)the existing curve between two end points, and playing with the control points to get a nice smooth curve,

Places where the curve pulls away from the edge can be resolved by moving the control point closer to the edge. In the above image the curve was found to have been pulled away from the edge. This was fixed by moving the control point a little to the left.

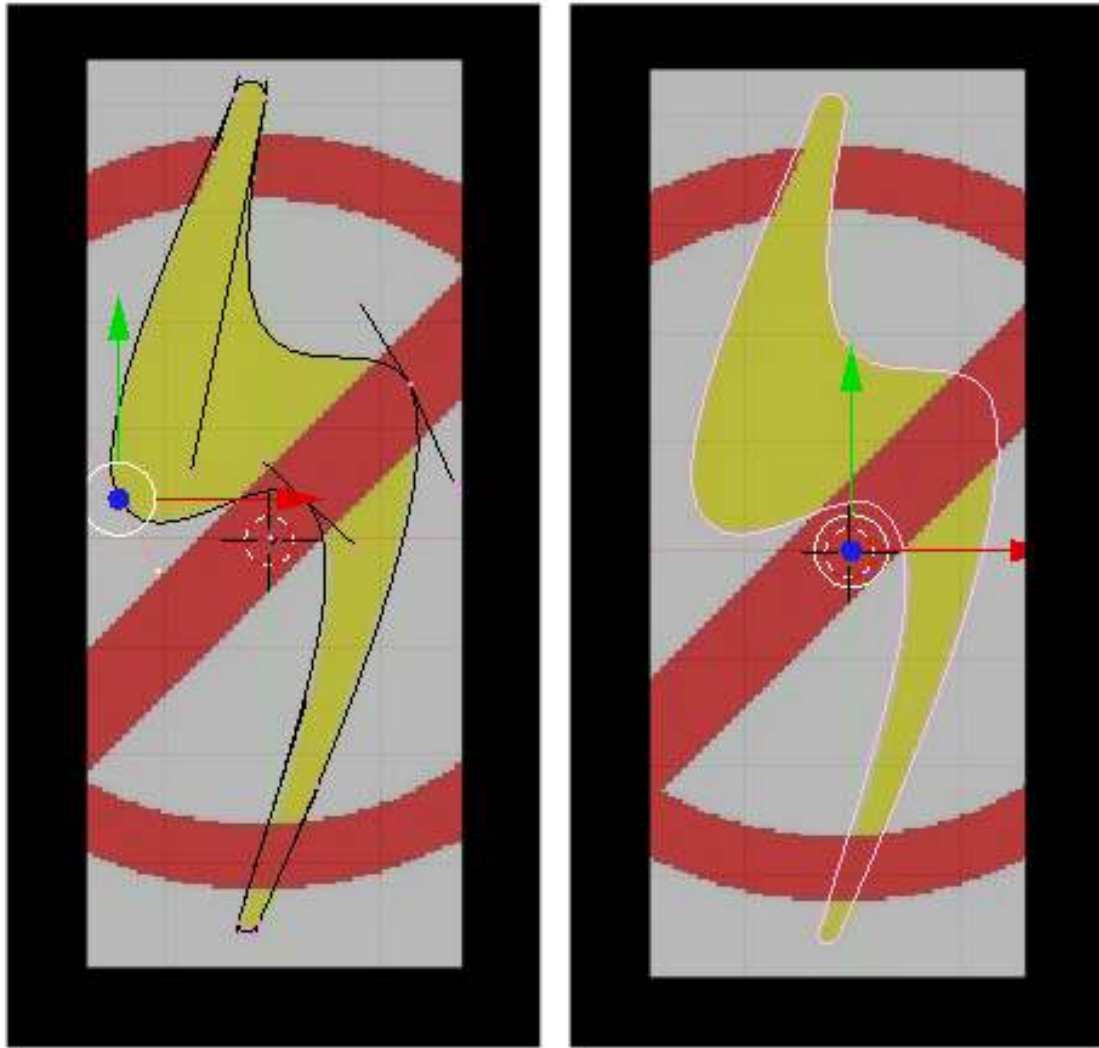


Figure 577 Final Polished Curve

Here's the final polished curve for my project. It is shown in both edit mode and object mode so you can clearly see both the control and end points on the left and the curve to the right.

Noob note: In blender 2.37 and later (not sure of earlier versions) pressing the **HKEY** toggles the control points between free and aligned (Edit Mode). Free Control Points are good for sharp angles, and aligned are good for smooth curves. This shortcut is in the **Space -> Edit -> Control Points** menu.

This concludes the tracing of the bolt. All that remains is making the curve 3 dimensional, applying a material and positioning the final object. Before doing that, we will trace the circle in the next part of the tutorial. Save this project if you want to take a break before continuing. You'll need it on the next tutorial.

66.5.3. Adding a Third Dimension

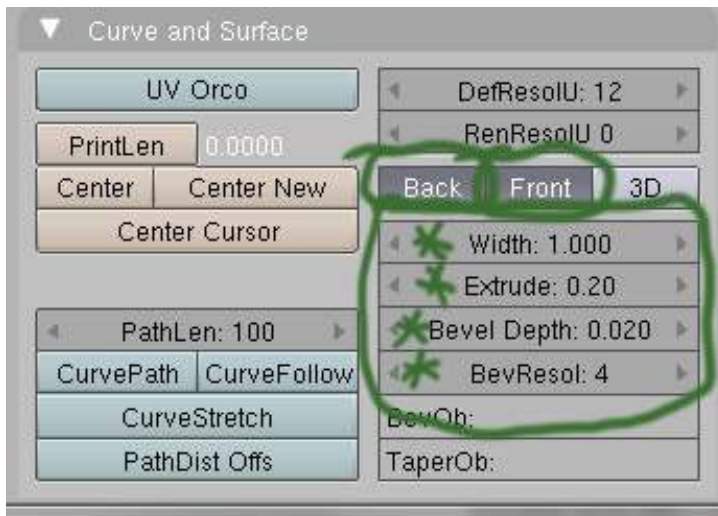


Figure 578

First, give the object some depth. if you are in wireframe, hit **ZKEY** to go in to solid mode, then down on the editbuttons screen, locate the "Curve and Surface" panel, and set the following values:

- Back and front buttons are selected (depressed), otherwise it will be see through.
- Extrude: 0.2 (the height of the extrusion on either side)
- Bevel Depth: 0.02 (the radius of the round bevel applied to the exruded edge)
- BevResol: 4 (the number of subdivisions on the bevel curve)

Noob note: In previous versions, Extrude and Bevel Depth were Ext1 and Ext2.)

Noob note: In versions 2.6x(possibly also 2.5x), go to *Properties panel --> Object Data --> Geometry* for these settings.

Now you can use your knowledge from earlier in this book to change the material and/or add texture to your logo. Feel free to rotate, add lighting, or whatever floats your boat. Don't forget to press **ZKEY** to toggle wireframe mode.

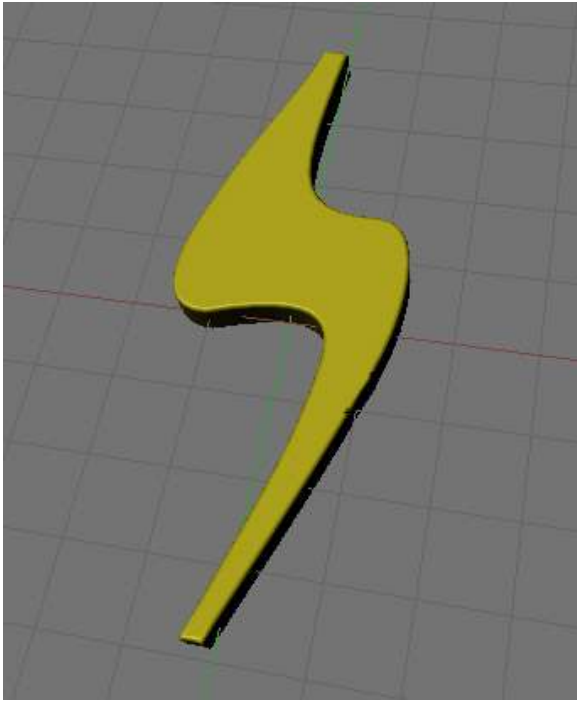


Figure 579

You should now have something that looks like the photo on the left.

66.6. Modeling the red circle

It will be helpful to make the lightning bolt distinct from the circle part of the logo, by applying a yellow material to it before continuing.

Method one is the easiest, so try it that way first.

66.6.1. Method 1 - using bezier circle

Outer circle

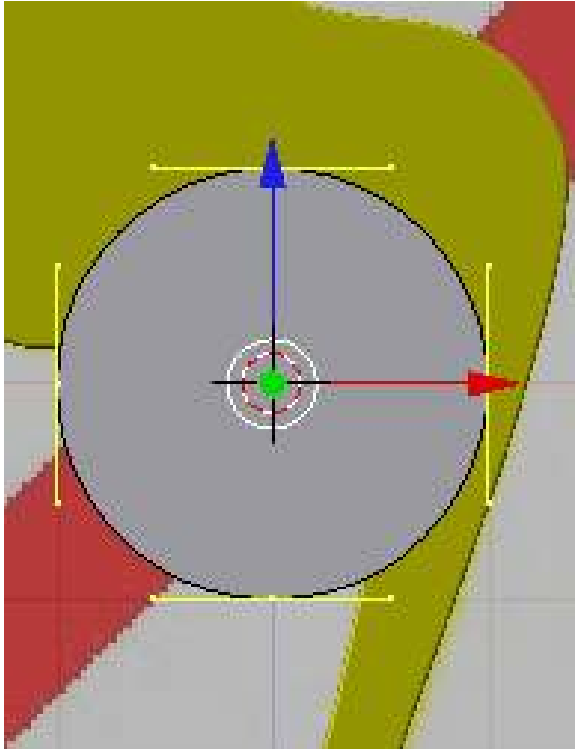


Figure 580 Bézier Circle

Switch to object mode by hitting **TAB** if you aren't already there. **LMB** somewhere near the center of the background photo. Press **space -> Add -> Curve -> Bézier Circle** to add a closed bezier curve with four points forming a circle. If you are in solid draw type, switch to wireframe with **ZKEY** so you can see the underlying image better. Hit **SKEY** to scale the bezier circle to fit over the circle in the image.

Noob note: You should stay in object mode for now, as it's easier to scale and move the bezier circle without having to keep track of which end points are selected.

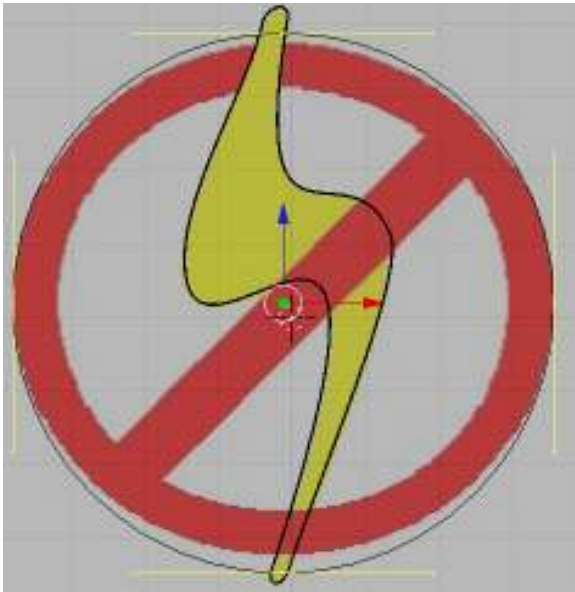


Figure 581 Match up left and right sides

You will probably find that the bezier circle is not dead center on the sample logo so you will need to move it with **GKEY** to center it. You may need to scale it and move it several times to get it right. You will also find that the circle in the sample image is actually a slight oval, so scale and position the bezier circle, along the X axis, and Y axis, so that it touches the circle in the image on the left and right sides.

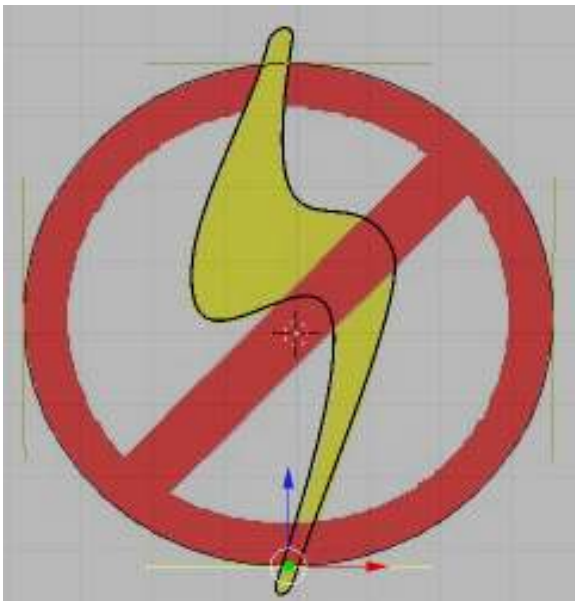


Figure 582 Move top and bottom points

Normally, you could then scale the circle and constrain it on the X and Y axis by using **SKEY**, but it turns out that the oval isn't regular anyway, so just select the point on the top and hit **GKEY** and then

XKEY, or **YKEY** to move it around until it touches the top of the oval in the image. Then do the same for the bottom point and you should have a pretty good fit.

Inner half circle 1

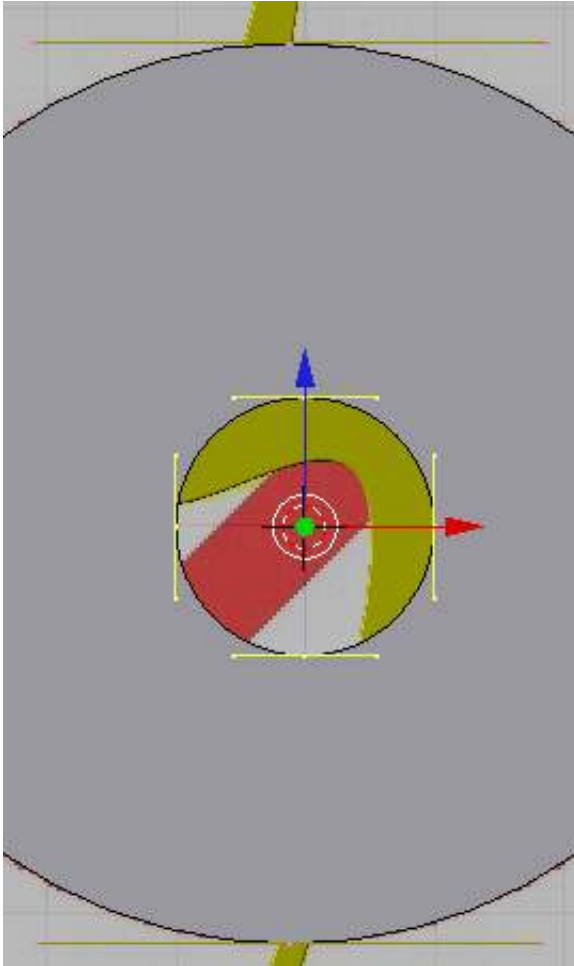


Figure 583 Circle in circle

Just to understand what's happening in the next steps, switch to solid view with **ZKEY**. As you can see, you now have a circle, but it's filled in the middle where you want to be able to see through it. To cut a hole out of the circle, be sure that the circle is selected, then hit **Shift+S** -> "cursor to selection, to put the cursor in the middle, switch into **edit mode**, if you're not already in it, and then **Space** -> **Add** -> **Bézier Circle**. A new circle will appear inside the larger circle. As you can see in solid mode, the new circle actually cuts its shape out of the larger circle surrounding it.

Noob note: make sure you are in Edit Mode or it will add the new Bézier circle without cutting it out of the existing one.

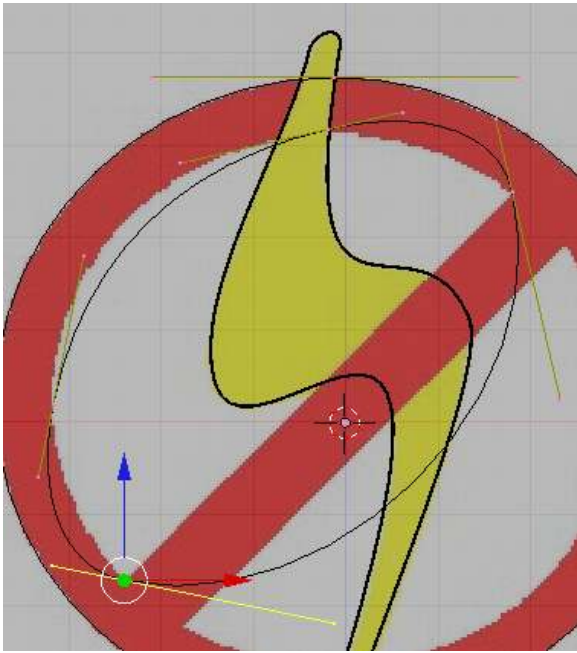


Figure 584 Place two points at either end of the bar

Switch to wireframe mode with **ZKEY** so that you can see the underlying image again. Scale up the smaller circle so that it approximately fits the inner part of the circle in the image. Don't worry about getting it exact since you'll be manually moving all four points anyway. Move the bottom point of the bezier circle to the top left corner of the bar that crosses the circle.

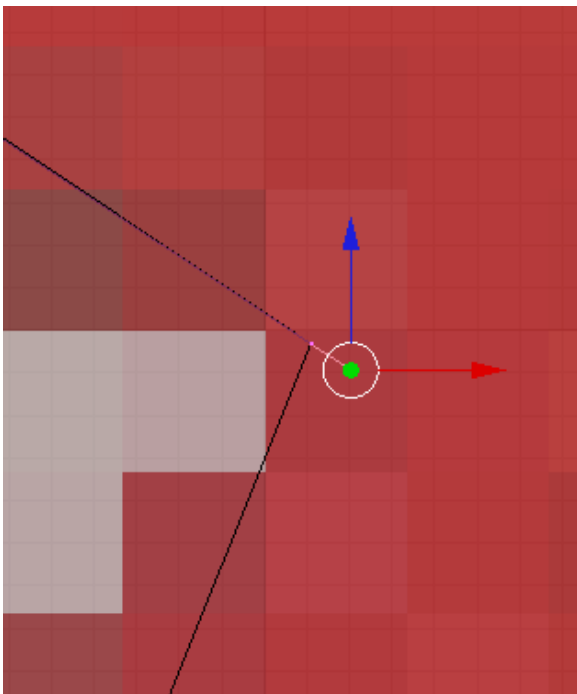


Figure 585 Place two points at either end of the bar

Move the right point of the circle to the other corner. When you create a Bézier Circle, Blender by default sets the alignment of all the control points to aligned. To make the diagonal bottom edge you need to break the alignment on the two lower sets of control points. Hitting the **HKEY** will toggle between free control points and aligned. Once you've selected the two lower bezier points and hit the **HKEY** to make them free you can move each of the inner control points to create a nice straight edge. Then move the other two points and adjust their control points until you have a pretty good approximation of the rest of the inside curve.

In Blender 2.63 and others, pressing the H key seems to hide vertices instead. To set control points to free, select the vertex, then in the curve tools under Handles, select Free. (can also be set via the menu at Curve-control points-set handle type-Free)

Inner half circle 2

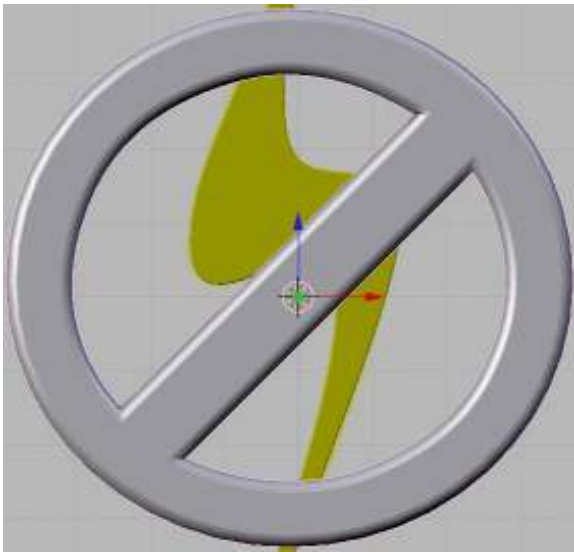
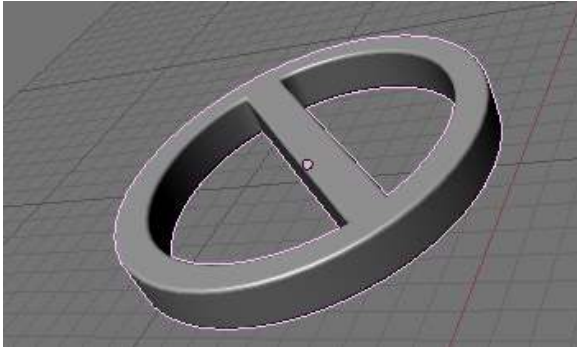


Figure 586 Wider bevel has a nice look

Next step is to press **space** -> **Add** -> **Bézier Circle** again and repeat the same steps, but for the lower opening in the logo. Once you've completed both openings, switch back to solid view with **ZKEY** and examine your work. Make any adjustments you need to swithing the draw type back and forth as needed.

Make it 3d**Figure 587**

The next step is to make this part three dimensional like you did with the lightning bolt. Go to object mode with **TAB**, then select the editing buttons.

Under curve and surface, set:

Extrude/Ext1 to 0.05,

Bevel Depth/Ext2 to 0.02

BevResol to 4.



Now that we have beveled it there will be a problem. Switch back to wireframe mode with **ZKEY** and you'll see that the bevel has widened everything so that the circle no longer matches the original image. This can be fixed fairly easily by reducing the width parameter under curve and surface to about 0.98.

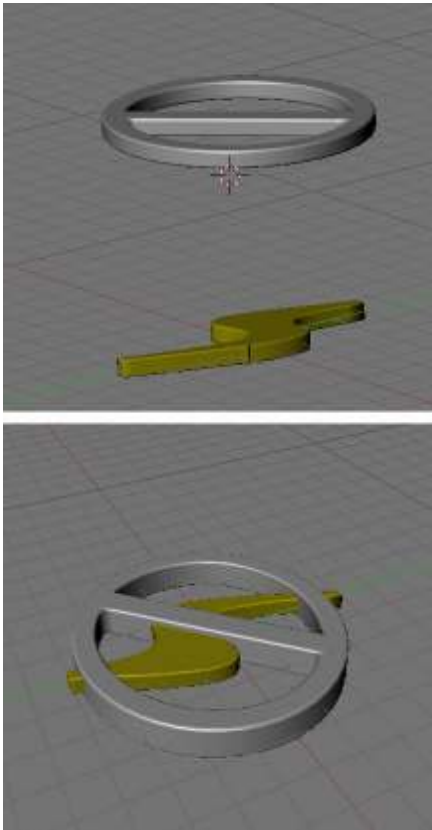


Figure 590

For final steps, select the lightning bolt again and switch into sideview with **NUM3** and hit **GKEY** and then **ZKEY** to move both elements up or down until the bolt is inside the circle.

Apply a red material to the circle and bar portion. Finally, you can go to view, then to background image and hit the background image button to hide the image now that it is no longer needed. At this point, you can add any finishing touches for lighting and camera angles and render the logo.

After rendering you will have something like this:



Figure 591 Final render

66.6.2. Method 2 - using mesh circle and edges

Instead of using the Bézier Curve, you can use Circles.

There are two ways of doing this:

Easy way

Place two vertices (CTRL+LMB), one on the inner circle and one on the outer circle so that they form a line. This line has to be perpendicular to the circle you are tracing. Then place the cursor on the middle of the circle. Then just use the spin tool under the Mesh tools tab (360 degrees and 32 steps). This creates a circle made of 32 adjacent squares. Give it more steps to increase the quality of the circle.

Detailed way

Go to the top view (NUM7), and press SPACE -> Mesh -> Circle. Accept the 32 vertices, you can make it less but it won't look as good.

Move it into the center of the circle, if you don't then I advise you have wireframe on for the moment (press Z). Then press S, for scale, and make it the correct size for the inside of the circle. Once you have that in the correct place, like so:

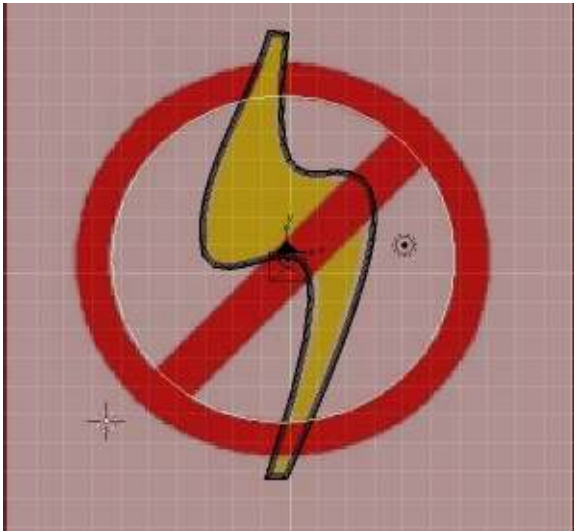


Figure 592 Adding the first circle

You may need to stretch it sideways a little, then press 'E' to extrude (choose Only Edges), press 'S' to scale and another sized circle will appear, size this appropriately then click LMB.

Deselect the second circle, then select four vertices that are near each other and that form a square.

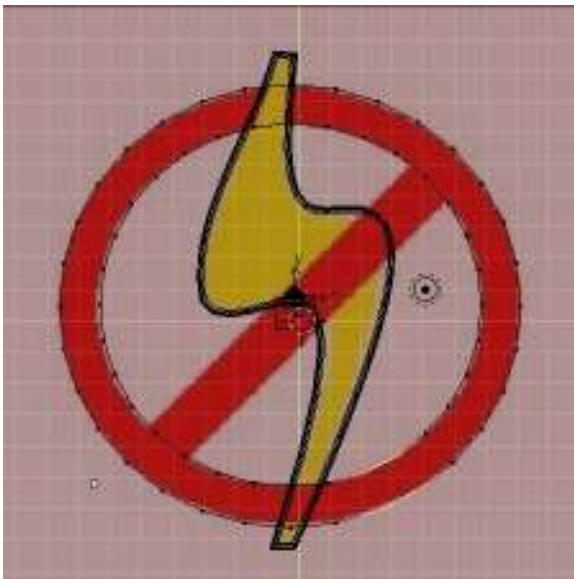


Figure 593 Selecting the four vertices

Now press F and a face will appear, i.e. the box will be filled (turn off Wireframe, press z). Now do this right around the circle. To do this, hold down Shift and Ctrl, then draw a circle around the two vertices you wish to deselect with the Left Mouse Button. Then draw a circle around the next two vertices while holding down Ctrl and LMB, **NOT SHIFT**. Shift changes the control from selecting, to deselecting.

Once you have gone all around the circle it's time to make the line through the middle. To make the crossing line you need to move 4 of the vertices slightly; example below:

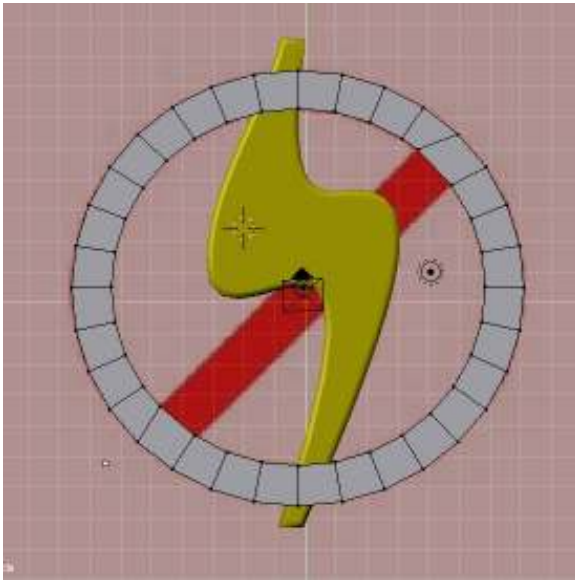


Figure 594 Moving the four vertices

Once you do this, highlight the 4 you moved, then press F.

Making the circle 3D

Highlight the full circle by pressing A either once or twice. Go to Side View, and press E for extrude and drag it down so that it is the same thickness as the lightning bolt (you'll see why).

Now look at what you have made... it looks nice enough but where the lightening bolt goes through the circle it looks a bit odd so we will make it look like the circle is laying on top of the bolt. Where the bolt goes through the circle, note the edges and the vertices. Move them so that the lines are just either side of where the bolt goes through. Then make new edges using CTRL+R on the outsides of these edges. Like so:

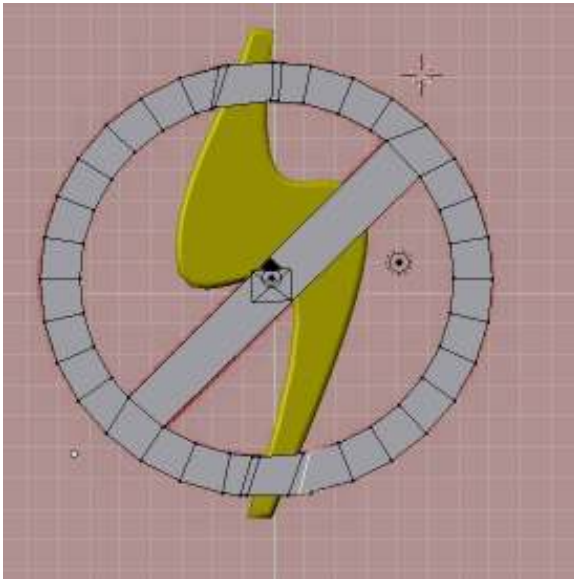


Figure 595 Creating the overlap

Now you have squares where the lightning bolt hits the circle. Change to Wireframe (z) if you are not already in it then highlight the 16 vertices of these boxes and raise them. Now do the same for where the center line crosses the bolt but create 4 lines instead of two. To explain why, I've created a diagram.

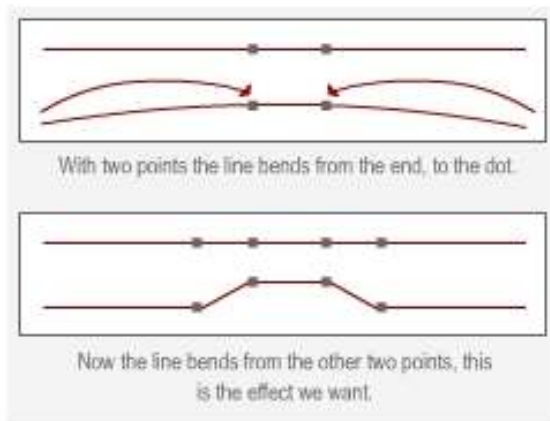


Figure 596 Why we use 4 points instead of 2

Now we can make the finishing touches, add subsurf and set it to 2 or 3 and add color! Then you are done :-)

I won't go over subsurf and adding color because people have covered that better than I could in previous 'Noob to Pro' pages. I think that's everything.

You should now be looking at something similar to:



Figure 597 The finished article

66.6.3. Method 3 - using mesh circle and a cube

Tracing the **No symbol** is somewhat more complex than the lightning bolt. The reason is this symbol is hollow and requires additional planning to trace than just following an outline of an object.

I'll be back to explain the difficulties in the near future, so stay tuned.

One way to do this is outlined below:

- The first thing to do is add a mesh circle. Hit **SPACE** and select 'Add -> Mesh -> Circle'. (This circle is mostly for measuring purposes, and will ultimately be removed.)
- Scale (**SKEY**) and move (**GKEY**) the circle so that it's sides are even with the sides of the logo circle, and the top and bottom are an even distance from the top and bottom of the logo circle. (The logo circle is not perfectly round.)

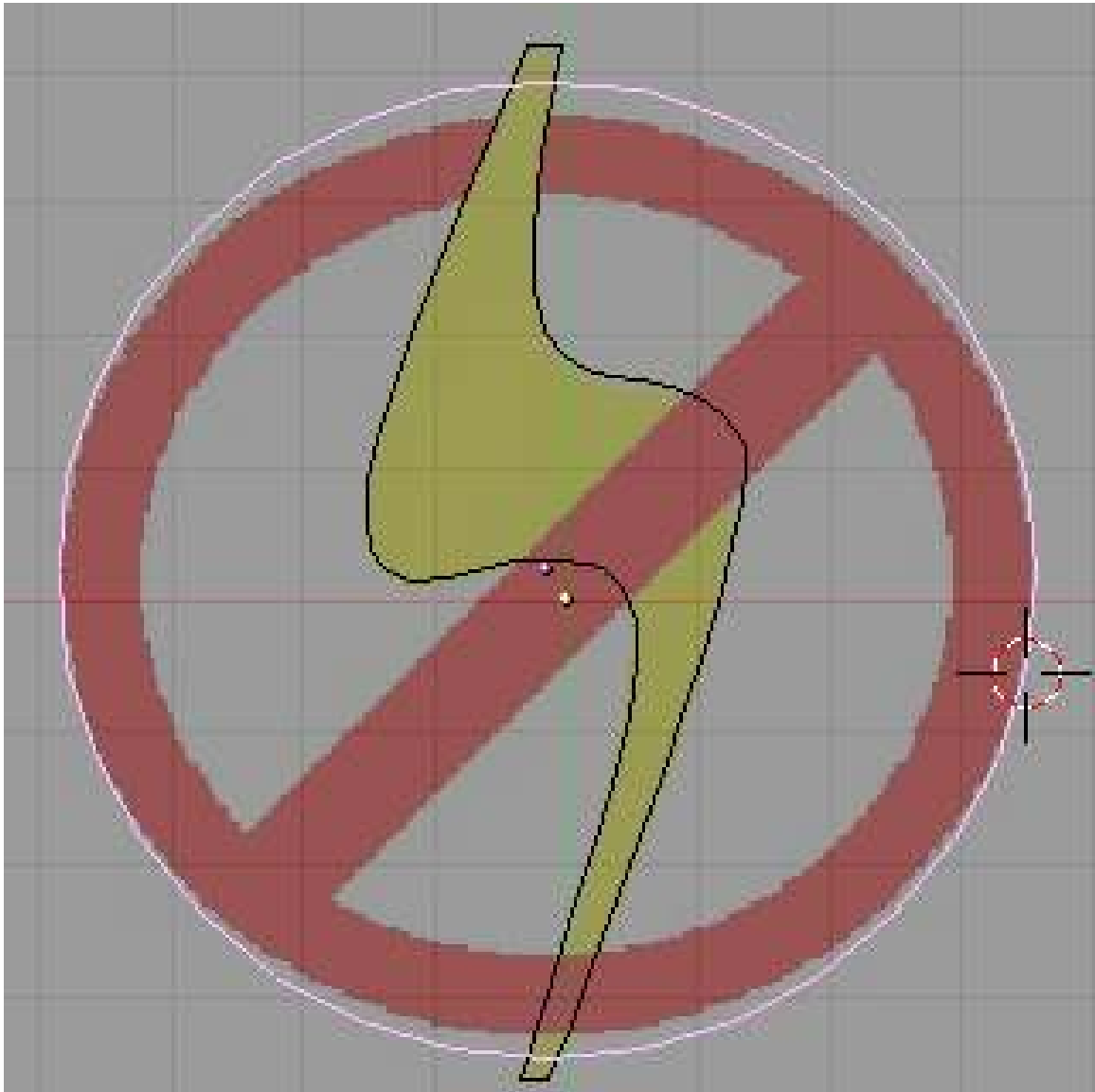


Figure 598

- Make sure you're in object mode (**TAB**), add a cube (**SPACE**, 'Add -> Mesh -> Cube') and place it (**GKEY**) on the right edge of the circle so that it is centered on the right edge.
- Scale (**SKEY**) the cube so that it is the same width as the wall of the circle and touching each side of the wall.
- Now, we want the 3Dcursor at the center of the logo circle. To do this, go back to object mode (**TAB**), and select the circle that we added earlier (**RMB**), then press **SHIFT+SKEY** and choose 'Cursor -> Selection'. (This is the main reason why we have this circle here.)
- Reselect the cube (**RMB**) and **TAB** to edit mode.
- Choose **NUM7** for the top view, and use **RMB** to select only the front face of the cube. (Before this you have to make sure that you are in 'Face select mode'.) In the image below, I've used **MMB** to rotate the scene slightly so that you can see that only **one** face is selected (Note: In this picture there are two faces selected. The one you should select is front one in front view **NUM1** or the bottom one in top view **NUM7** - it's the same face).

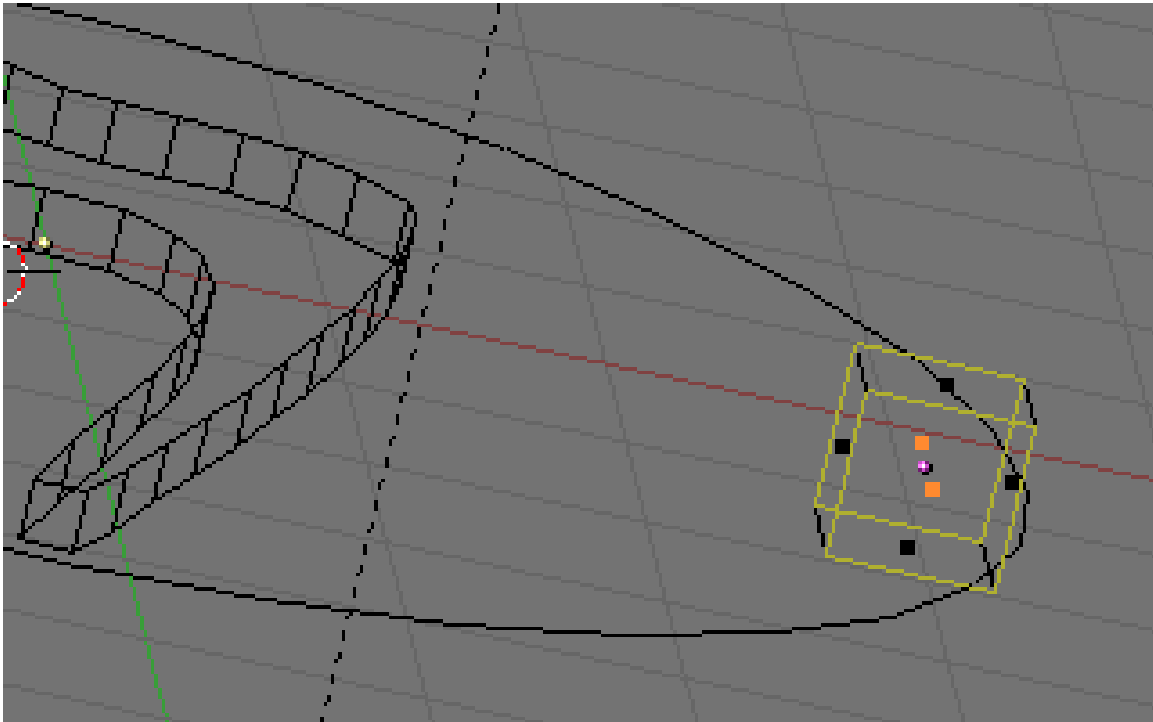


Figure 599

- Press **XKEY** and choose Erase Vertices. This will reduce the cube to a square

(User note: Erase Vertices will erase all the vertices connected to the two faces selected. In this case this is all the vertices in the cube and will completely remove the cube. I think "Erase Faces" would be correct.) (Another User note: Only one face shall be selected, not two as shown in the picture.)

- Press **NUM7** to go to the top view then select the Editing panel (**F9**).
- Press **AKEY** to select all, and then make sure that the rotate settings are set to degr:360 , Steps:50 , Turns: 1. (Actually, Steps can be whatever number suits your fancy. You may want to play with various values.) This is just like we did for creating the man's hat.
- Click on 'Spin' to extrude the square into a ring.

Next we will create the bar.

- Staying in Edit mode, hit **SPACE** and select 'Add -> Cube'. If you haven't moved the cursor since the ring extrusion, it should appear in the middle of the ring.
- Expand (**SKEY**) the cube so that it just encompasses the inside circle of the ring

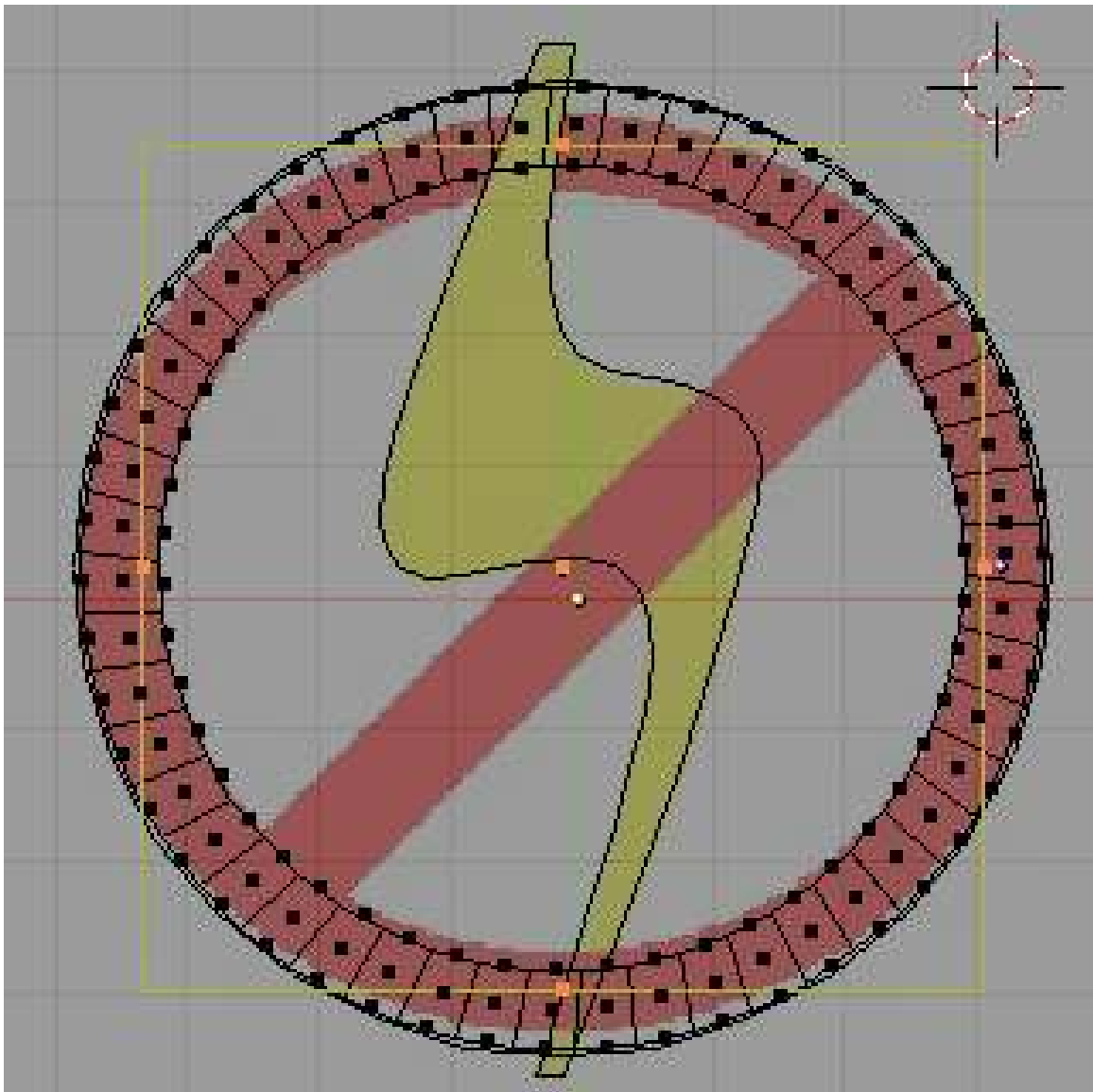


Figure 600

- Scale (**SKEY**) the square in the Y axis (**YKEY**) so that it is approximately the thickness of the logo bar.
- Return to object mode (**TAB**), and rotate (**RKEY**) the ring together with the just designed bar so that it is parallel to the logo bar.
- In Edit Mode (**TAB**), squash the ring in the Y direction (**SKEY**, **YKEY**) so that it fits the logo ring.

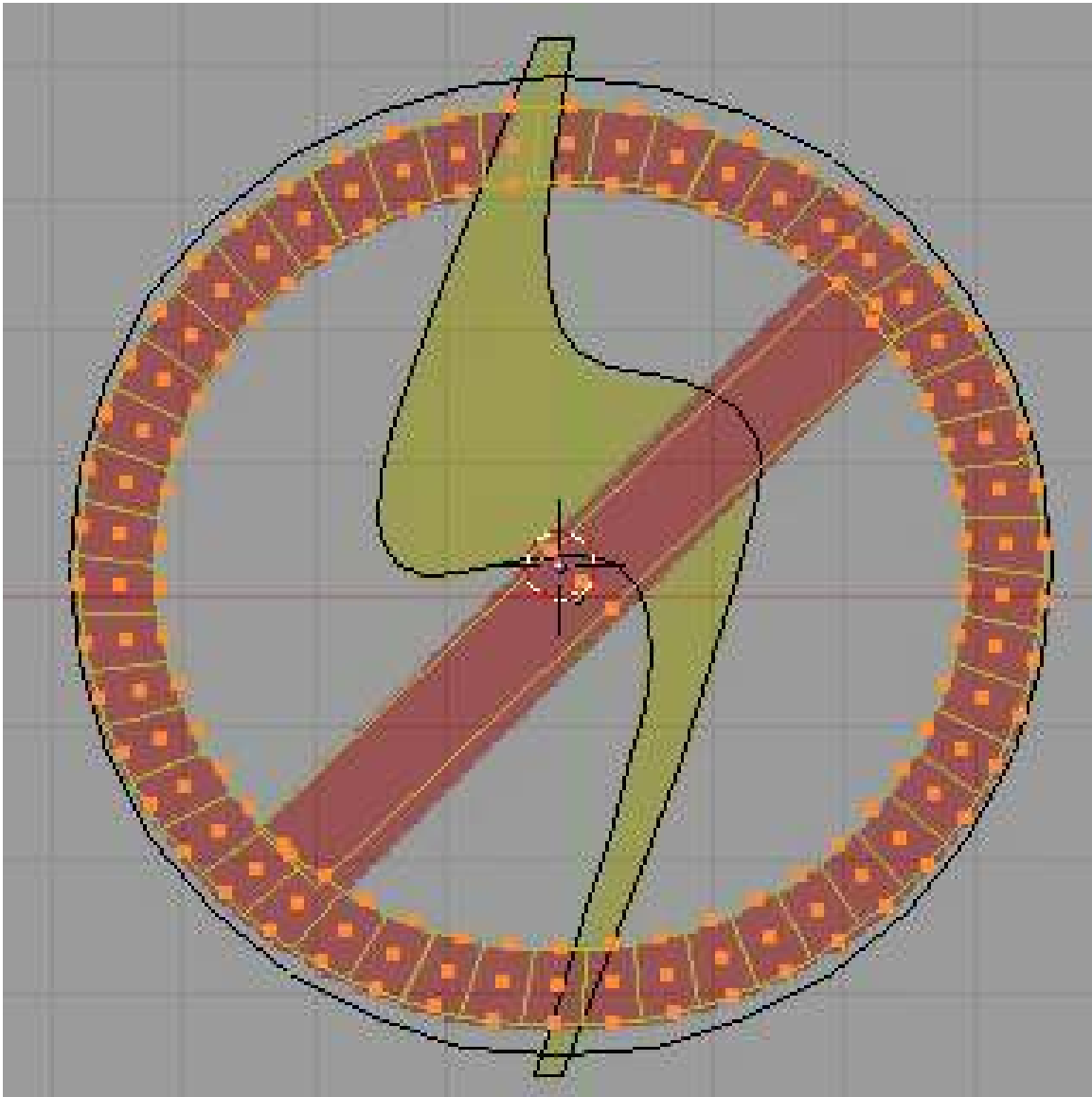


Figure 601

- In edit mode, reselect just the bar. To do this, choose 'Face select mode', and do an area select of the face selectors near the center of the bar (use the **BKEY**, or **SHIFT+RMB** methods).
- Re-rotate (**RKEY**) the bar (in Edit Mode) so that it is parallel to, and in the middle of, the logo bar.

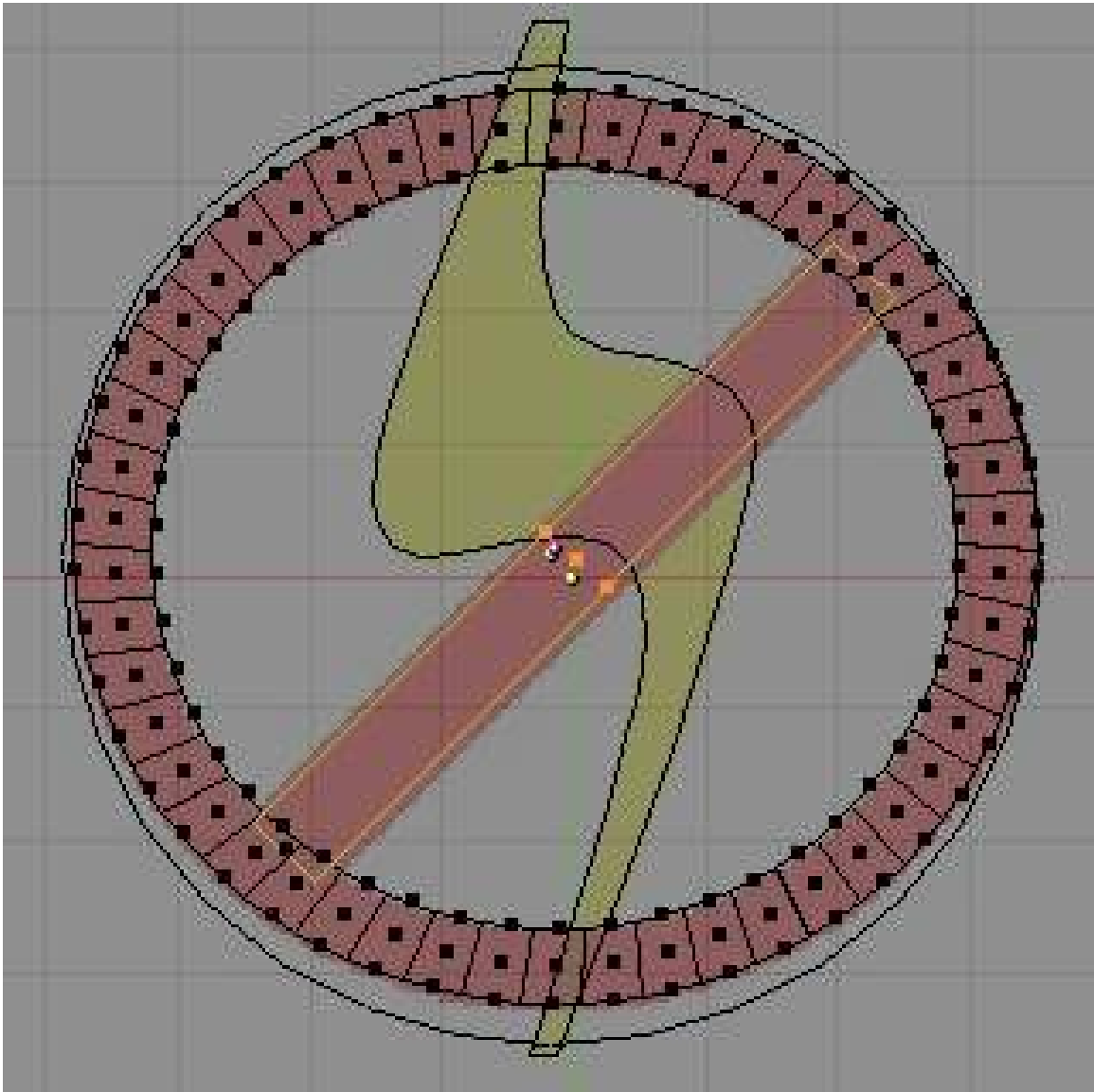


Figure 602

- The next thing we need to do is scale the bar so that it is precisely the same width as the logo bar.
- To do this, choose scale (**SKEY**), and press **YKEY** twice. This will go to local scale mode (local to the object). This is why we were rotating the entire ring, since the ring and bar are part of the same object.

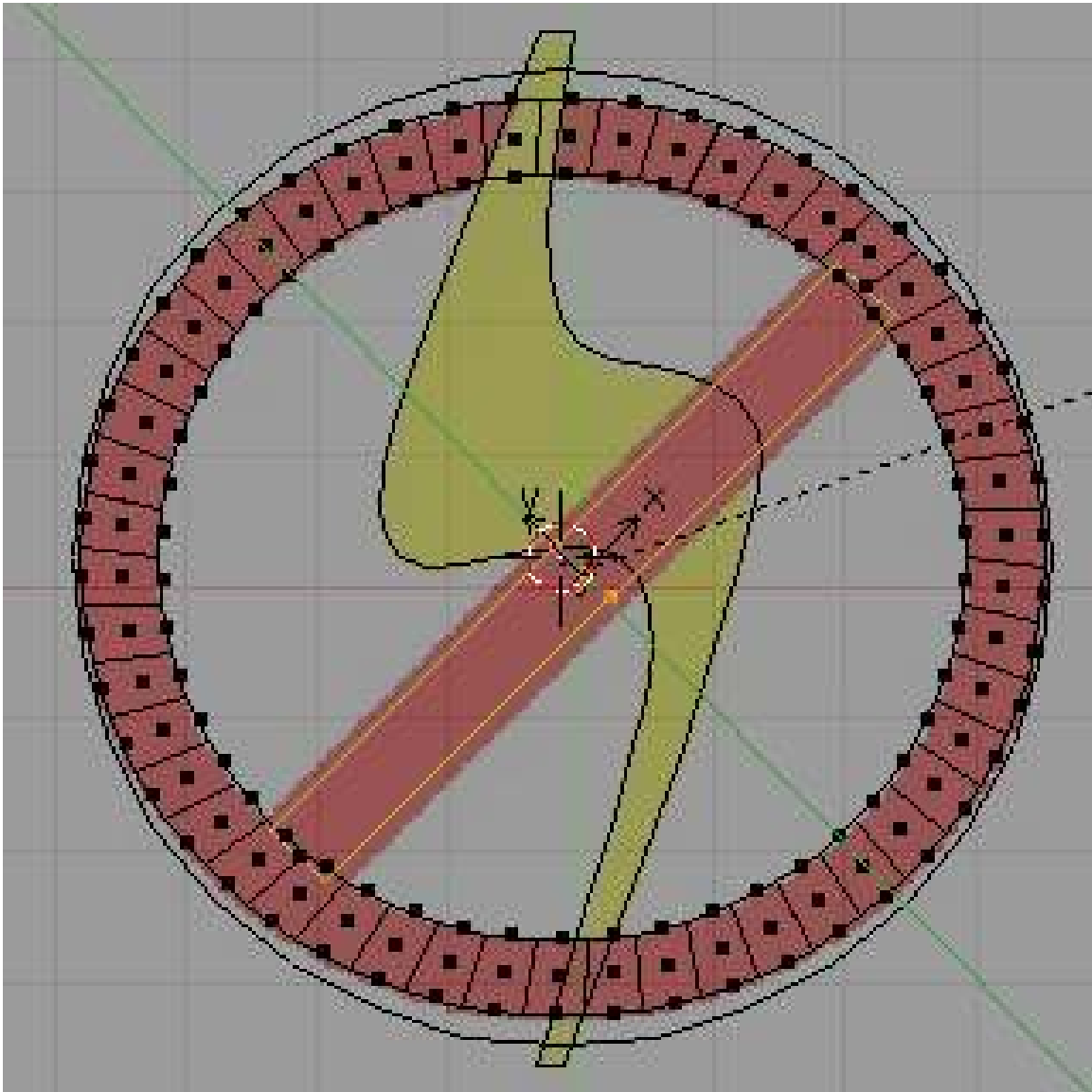


Figure 603

- We no longer need the original circle now, (it should be sticking out from the top and bottom of the ring), so select it (**RMB**), and delete it (**XKEY**).
- If you hit **NUM3** You'll see that the bar and the ring are at completely different heights than the lightning bolt. Select and scale them so that they are all roughly the same height (the bar should be slightly higher than the bolt, and the bolt slightly higher than the ring so that all the proper parts are covered.).
- You can use the trick above to select (**RMB**) only the bar, scale it in the Z axis (**SKEY**, **ZKEY**) and then hit **SPACE** and choose 'Select -> Inverse' to select the ring (all but the bar).
- When done rescaling, press **NUM7** to look at your handiwork.

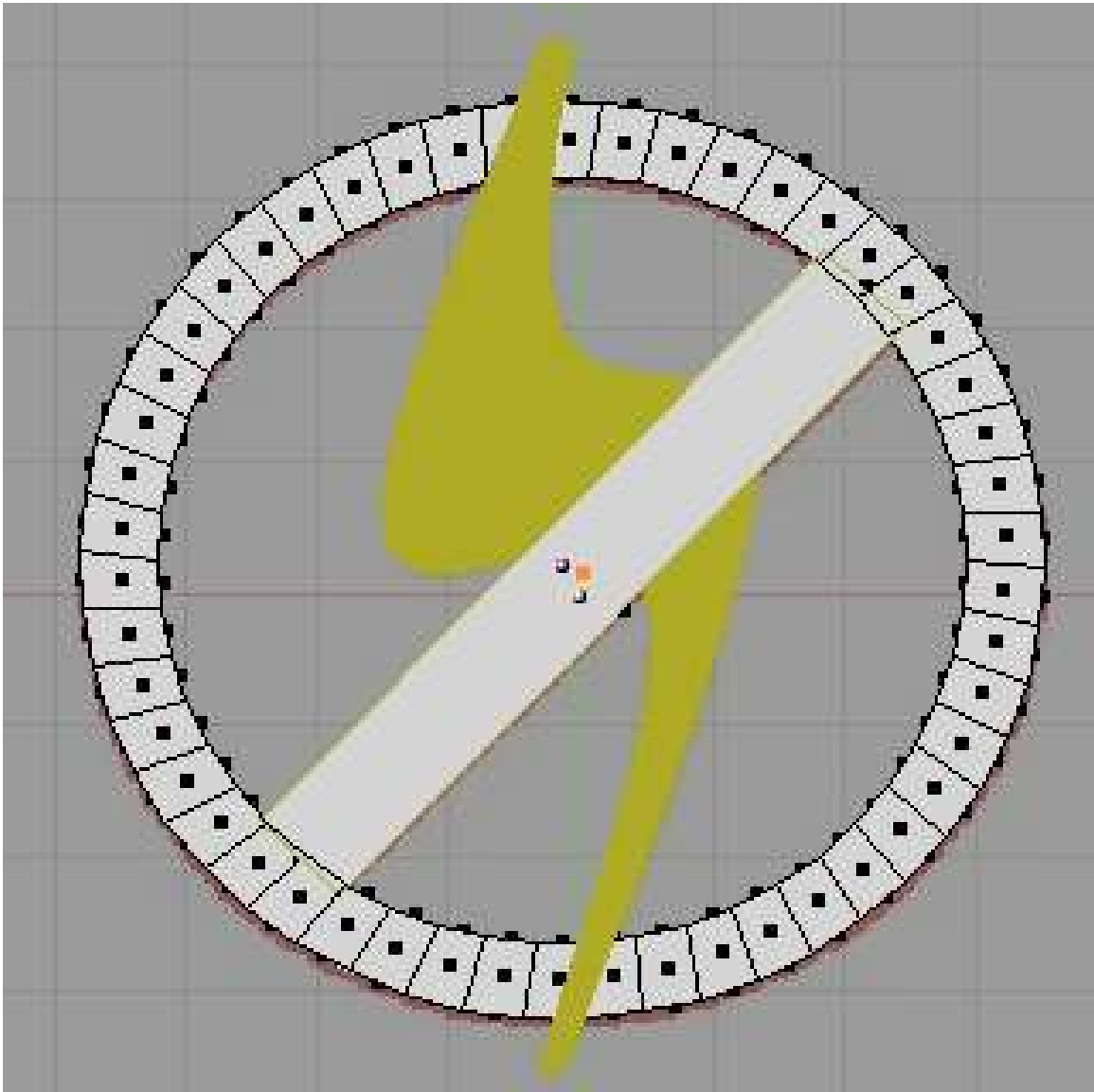


Figure 604

- There is one last bit, which is to slightly rotate the whole ring around the X axis so that it is below the top of the bolt and above the bottom.
- Once that's done, all that's left to do, is color in the ring.

66.7. Extra Practice

An Alternate More Difficult Tutorial³

³ <http://forum.nystic.com/viewtopic.php?f=34&t=3025>

67. Curve and Path Modeling

Frighteningly enough, we know what you're thinking. You're thinking that mesh modeling is cool and all, but what about if you want to make an object that has smooth curves in it? Ok, so maybe you weren't thinking that, but in case you're curious move on to the next page to learn more.

68. Intro to Bezier Curves

68.1. Bezier curve basics

You wanted curved lines? Well here is how to get them, using Bezier curves¹.

- First start a new Blender project, and delete the default cube.

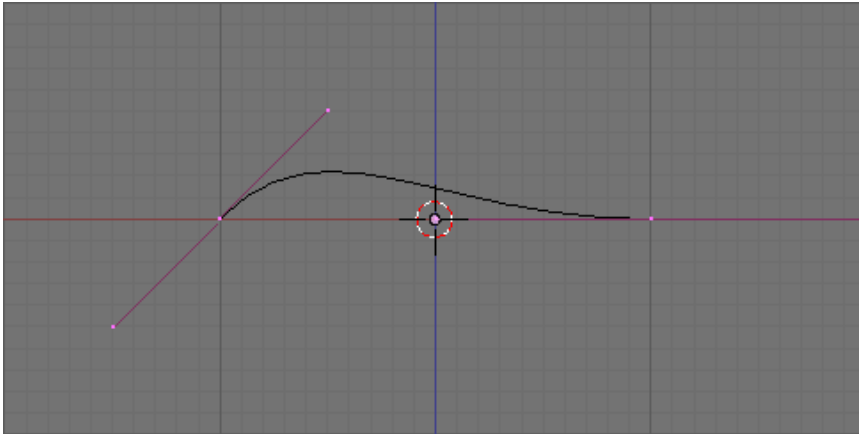


Figure 605 It should look like this.

- Press: 'SPACE -> Add -> Curve -> Bezier Curve' to create a new curve.

As you can see, there is a black, curvy curve on your screen, inside what looks like a normal mesh (purple).

There are 2 parts you can move on a bezier curve.

- Control Point - Define where each section of the curve starts and finishes.
- Handle Point - Tells the curve how to bend and turn before getting to the destination.

By now you should have discovered you can drag the points around using the **GKEY**, and that the handles are fixed in position relative to their control point. This is not the only thing you can do.

Here is a list of things you can do to the curve to modify it:

1. **Grabbing:** You can drag the points around separately or in groups.
2. **Rotating:** Using the **RKEY**, you can rotate a control point around with your mouse. This is different from a mesh, as rotating the control point will move its handles too.
3. **Scaling:** Using the **SKEY**, you can scale either an individual handle, or a whole control point at once. This changes how strongly a handle influences the curve's shape, not its direction.

¹ http://en.wikipedia.org/wiki/Bezier_curve

Whatever you do to the defining points, Blender calculates the curve so it flows smoothly from one end to the other. When you leave Edit mode, only the black curve will remain visible, so don't worry about your control points getting cluttered!

Noob note: At any time you can convert a Bezier curve into a mesh by entering object mode (**tab** if in edit mode), hitting **ALT-CKEY** and selecting 'mesh' from the menu. You may then operate on the mesh like you would any other. This is great for creating a smooth Bezeir curve, converting it to a mesh, and spinning it around an axis.

68.1.1. Extending your curve

So far your curve has only two control points, but in order to build up more complex shapes you will need to add many more! There are quite a few ways to do this, but lets start with the basics:

Extruding

Much like a mesh, you could use **EKEY** to extrude one end of your curve with a new control point. Select an end control point (not a handle, this will cause you a few problems) and press **EKEY**, then **LMB** when it is in the correct location. The new point will have an exact copy of the handles of your old point, so you will have to rotate and adjust it accordingly.

Noob note: you can use **CTRL+LMB** to achieve the same effect.

Subdividing

Instead of adding onto the end of you curve, you can also split up sections of it into more control points, and here's how:

1. Select two adjacent control points using **RMB**.
 2. Press the **WKEY**, and select **Subdivide** from the popup menu.
- The segment should now have a new control point mid-way between the first two, that sits exactly on the curve that already exists. Now you can add more shape to this section using the new point.

68.1.2. Filling your curve

So far we have been working with a continuous curve, with a definite beginning and end. But what you can do is turn this wiggly line into a wiggly shape.

- Convert the shape from 3D to 2D in the Object Data panel (near Material). Older versions of Blender defaulted to 2D.
- Select any part of the curve in Edit mode, and press the **ALT + CKEY**. This will connect your last control point back to the first, and fill in the shape.
- If you cannot see any fill, you are probably in wire-frame mode. Press the **ZKEY** to recover.

- If you have weird results, say fill leaking out from the curve or empty sections, you have probably got a couple of parts of the curve crossing themselves. **Curve sections should never cross if you are making them into a filled shape!**

68.1.3. Handle types

Thus far with curves, whenever you move a handle around, its partner will move too to make the curve flow smoothly through the control point. This is not the only way to do things however, so here is a list of the different things you can set them to do:

1. **Aligned / Linked rotation** (Pink handles): Both handles always point in opposite directions. This leads to a smooth curve.
 2. **Free rotation** (Black handles): The handles can be moved independently of each other, you can form corners and sharp bends in the curve.
 3. **Automatic** (Yellow handles): The handles are linked in rotation, and automatically rotate and scale to form the nicest (mathematically speaking) smooth curve through your points. If you try to move a handle, both will default back to Linked rotation.
 4. **Vector** (Green handles): The handles act as vectors, with each pointing directly towards the next control point. This leads to completely straight edges (assuming the next control point is also a vector), so you can construct polygons. If you move a vector handle, it will default to Free rotation, but the other will stay as a vector.
- Note, that the hotkey for handle type is **VKEY**, the specific window should occur, then chose from the list.
 - Note, each handle from a control point can be assigned a separate type independently (excluding Auto). Just select only the handle vertex instead of the control vertex.

68.1.4. Combining curves

As well as having one single filling curve in your object, you can combine several unconnected curves to create more complex shapes.

1. Firstly create a filled shape, as described earlier.
2. Now, stay in Edit mode and 'SPACE - Add - Curve - Bezier Curve' to create a new curve on top of your existing one.
3. Manipulate the new curve so it fits inside the larger shape somehow, this will not work if the lines intersect.
4. Making sure only the new curve is selected, fill it with **ALT+CKEY**.

You should see that the space enclosed by the inner curve is no longer filled. You can continue by adding more curves inside a larger one, or even a larger one around them all.

68.2. Extruding from 2d to 3d

You may have noticed you can only modify the curve in two dimensions, and now it's time to explore the third dimension! Extruding is where you define a two dimensional 'profile' shape, and it is 'swept' through space to create a volume.

Noob note Bezier circles are not true circles, they are approximations. For artistic purposes, this may not really matter, but for precision modelling only NURBS circles should be used. This is due to the math used to describe the Bezier and NURBS curves.

68.2.1. Make a simple face of bezier curves

- **Make a curve and fill it:** Use the steps above to create something simple, and fill it in using **ALT+CKEY**.

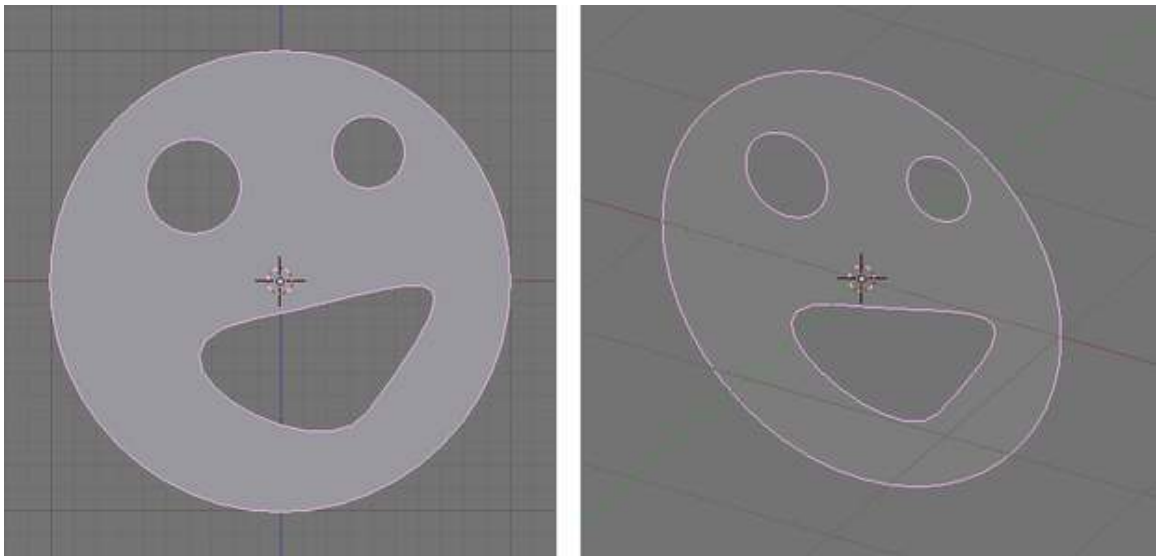


Figure 606

Noob note: you may find it easier to use Bezier circles, instead of filling discrete curves.

68.2.2. Extrude the simple face

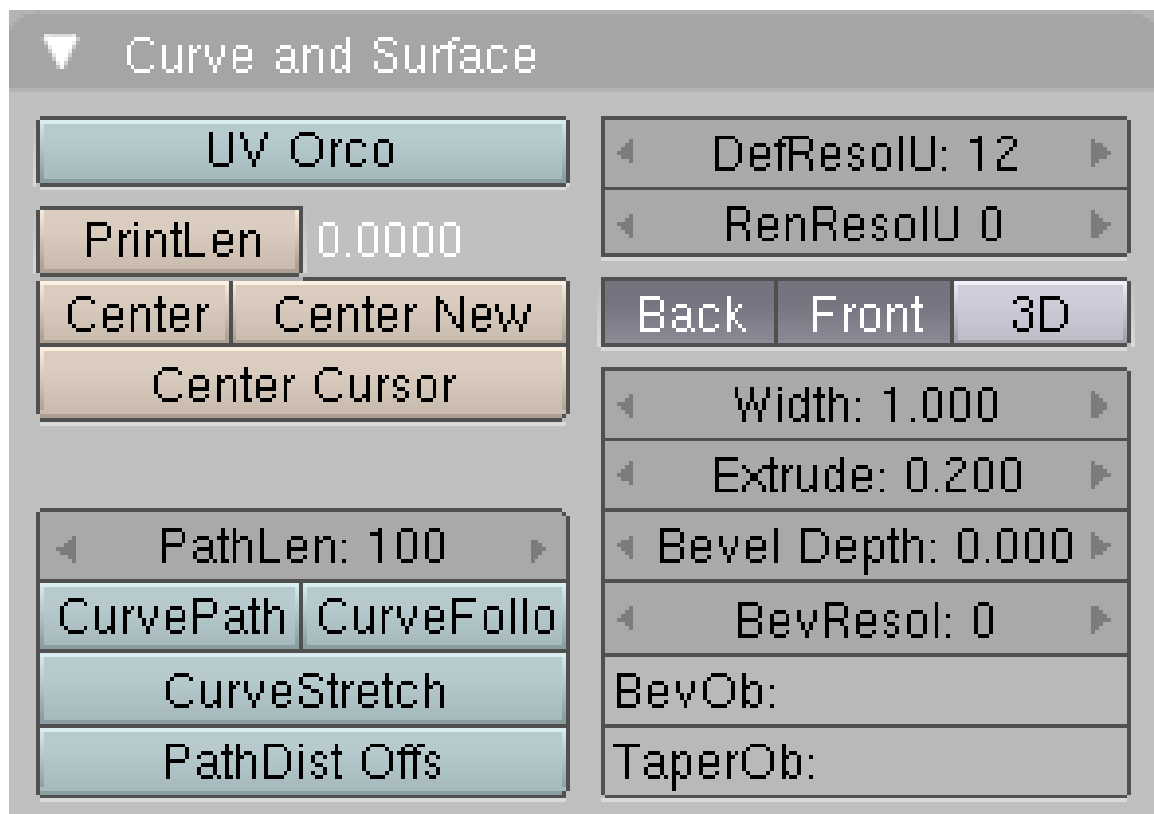


Figure 607

Set the extrude depth: Click on the Object Data button and find the Geometry panel. (Older versions: find the 'Curve and Surface' box in the Editing tab of the Buttons window.) There is a slider called Extrude. Set the Extrude depth to something other than 0, and probably less than 1.



Figure 608

Behold, your 2d curve has transformed into a neat 3d structure. The great thing is, you can still edit the curve as if it were just 2d, and the changes will update in real time.

68.2.3. Bevel the simple face

Now you have an extruded shape, you should start playing around with some of the other curve settings on offer, so here is a description of how the Bevel depth and Bevel resolution sliders work.

Try setting the Bevel depth to a small value, say 0.02. This will cut off all of the sharp edges, and give a bevelled effect all around the shape.

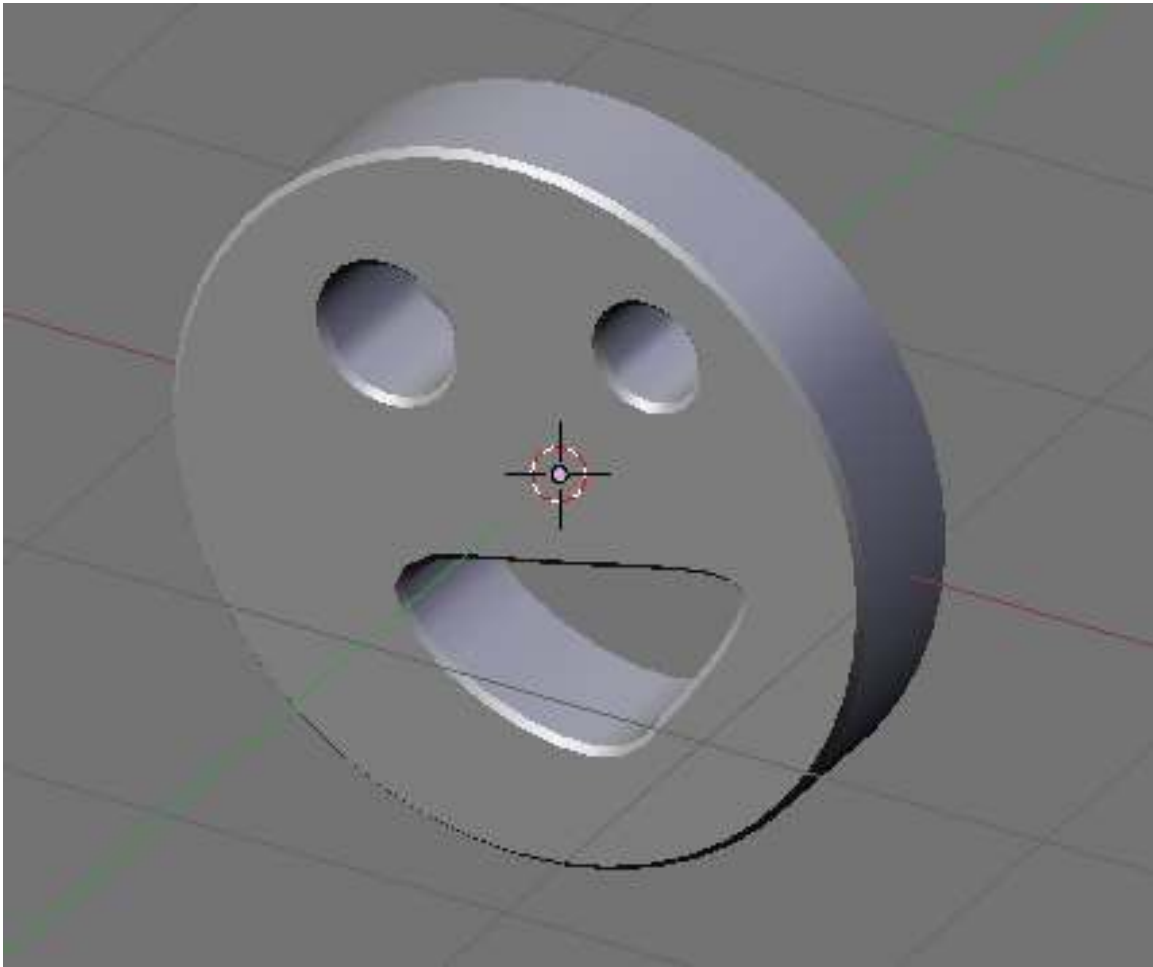


Figure 609

As you may guess, Bevel resolution decides how many times the algorithm divides up an edge. Higher values than 0 result in smooth curves rather than sharp edges, but dramatically increase the number of vertices in the shape.

Try setting the resolution to 3 or 4, you should see an effect like this:

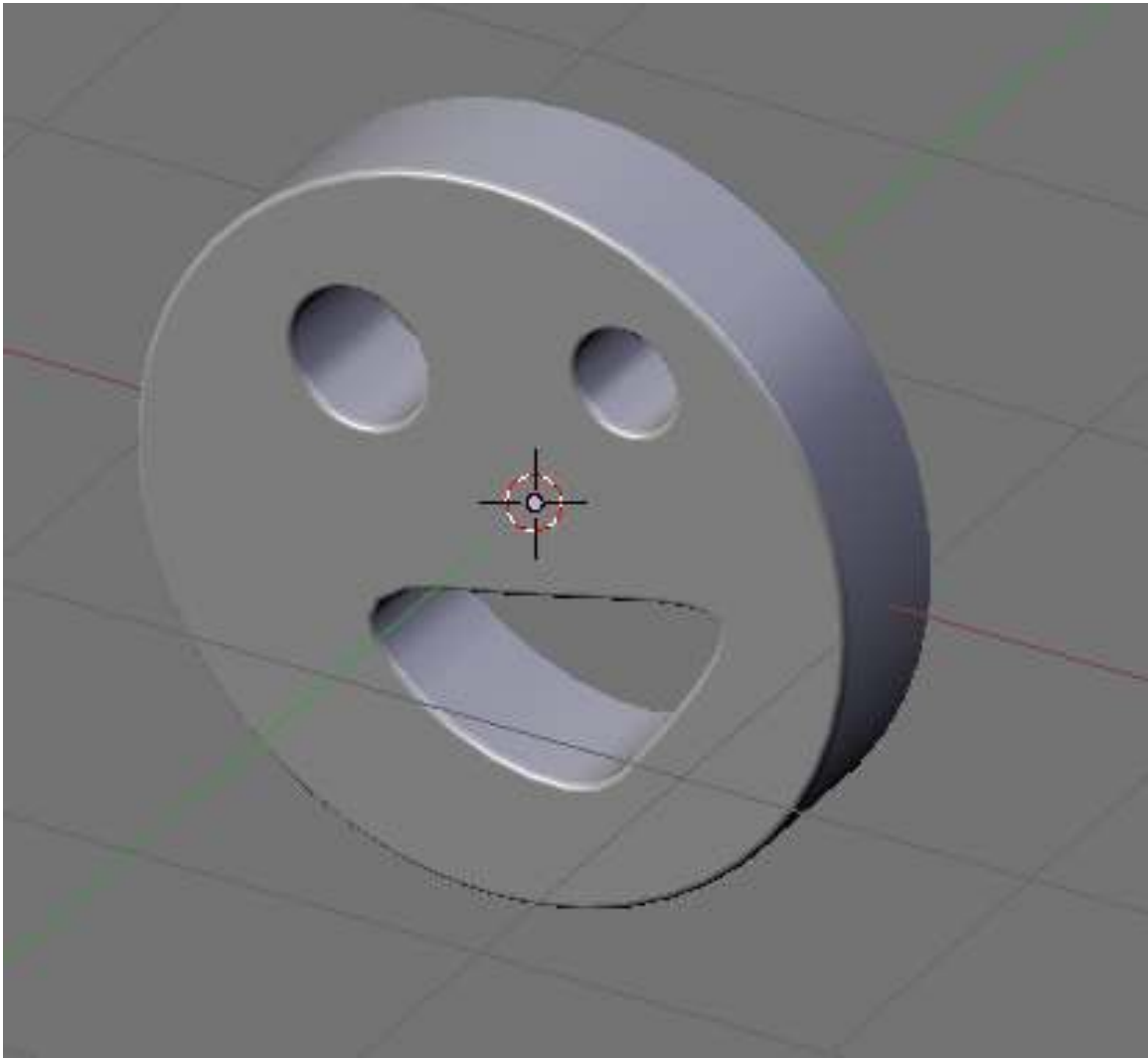


Figure 610

68.2.4. Using a bevel object

Noob note: *After you finish making the filled shape exit to object mode.* Then add a curve and rename it to bevel or anything under the Link and Materials tap or just use the given name of the new curve. Then deselect the new curve and select your filled shape. With the filled shape selected type in the name of the new curve that you have just created in the BevOb: box. It is under the Curve and Surface tap at the lower right corner.

This technique allows for much more sophisticated extending of our curves into 3d. Basically, we can create a new curve which will be extruded all the way around our main shape.

1. **Create a curve:** Once again, make a filled shape like our friend the smiley. **Noob Translation:** In Object Mode, Add a bezier curve, go to edit mode and make a smiley.

2. **Create a bevel object:** Add a new curve and return to Edit mode, leaving it as the default shape. **Noob Translation:** Go back to object mode and add a new bezier curve (preferably a closed curve using **CKEY**), then go to edit mode.
3. **Assign the bevel to the curve:** Type the name of the second curve (Default is Curve.001, or just Curve if you used a circle before) into the relevant box in the 'Curve and Surface' menu. **Noob Translation:** In edit mode (editing the second bezier curve) under the links and materials tab, on the OB: text field, change the name of the curve to bevel (the name doesn't matter though, it just makes it easy to remember. Go back to object mode and select the smiley and go to edit mode. Under the Curve and surface tab on the BevOb: text field write bevel (or the name you chose for the bevel line).

Noob note: DON'T PANIC if you end up with a useless mess! This is expected, as we have yet to understand how the bevel object works.

Noob Note Took me the better part of a morning to work this out. Here is a video tutorial to help anyone else finding it hard to understand <http://www.youtube.com/watch?v=m5b43M62Hq4&feature=plcp>² Hope it helps.

Basically, the curve we created as a Bevel object is used to define the shape of the edge around our filled shape.

- The height of each part of the Bevel object above or below the local X axis is how far in front or behind the filled shape to extrude from the flat curve.
- The distance to the left or right on the local Y axis is how far towards or away from the center of the shape to pull each bit of the curve.

For our purposes, we want very little variation in the width of the bevel. This will hopefully solve the problem of all sorts of bits of the shape overlapping each other.

The default curve also starts and stops on the x-axis, we want to move the start and end points so one is higher than the other. (Preferably either side of the x-axis too, we don't want the extruded shape too far away from its curve).

Assuming you have kept the default new curve for your bevel object:

1. Move the left-most point down a whole large grid-square, and right until it is two or three small squares from the main axes.
2. Move the other point up a whole grid square, and left until it is roughly on the blue center-line.
3. Modify the control handles if you feel it's necessary.

This should leave your shape looking nice and neat, extruded quite a long way with a smooth curving edge. Experiment with the shape, see what works and what doesn't!

Noob note: if you have trouble cleaning up the 'useless mess' after assigning the bevel object to the extruded object, then keep the bevel object close to its central point (not sure what this is actually called, but it is the pink dot that you see when you select the curve). The following steps helped me resolve the 'useless mess'.

1. If you are in object mode, change to edit mode, then type A to select the entire curve (you may need to type A twice to first deselect, and then select the curve).

² <http://www.youtube.com/watch?v=m5b43M62Hq4&feature=plcp>

2. Now type G to grab and move the curve, move the selected curve over the pink dot that you see, and you should start seeing your extruded object look more like it was before it became the 'useless mess'.
3. It may still need some work, so rotate (R) and scale (S) the bevel curve until the edges of your extruded curve match the shape of the bevel curve.

Noob note: When rotating the bevel curve, you might notice that you can easily invert the bevel by rotating it by 180 degrees.

Noob note: The sliders for Extrude, Bevel depth and Bevel resolution are not used once a bevel object is in place.

(To come: Closed shape Bevel Objects)

68.2.5. Front and Back

I would use wood working analogies to communicate what I expected to see and what I saw instead. Using a bevel object I assumed would be very much like using a Router with an Edge molding bit attached. When you run your wood piece thru your router its edge gets shaped/hollowed out. All those nice decorative edge effects, we see on furniture, and joinery aids, we may not even know exist, are generated by routers.

With that hope in mind I created a bevel curve object making sure it was smaller than the Extrusion depth (about 1/5th of it), it was an L looking curve, hoping it will hollow out my Smiley's edges. What I am seeing instead is quite surprising: The bottom 4/5th of my first filled extruded curve (my Smiley equivalent) is removed, top 1/5th remains. As if only the top surface has remained after all that work.

68.3. Create a filled, extruded shape

Follow the process above, for speed just use an extrude depth.

Disable Front and Back: Click both 'Front' and 'Back' in the 'Curve and Surface' menu to disable them.

Now you should be able to see right through the curve! Cool isn't it.

Noob note: You can disable just one of the faces and leave the other intact if you like.

69. Curves In 3d

69.1. Bezier Curves in 3d

In the previous page, we learned how to create and basically manipulate Bézier curves in two dimensions, as well as extruding them into the third. Here we will be exploring the '3D' button you may have been wondering about in the 'Curve and Surface' menu and some of its implications.

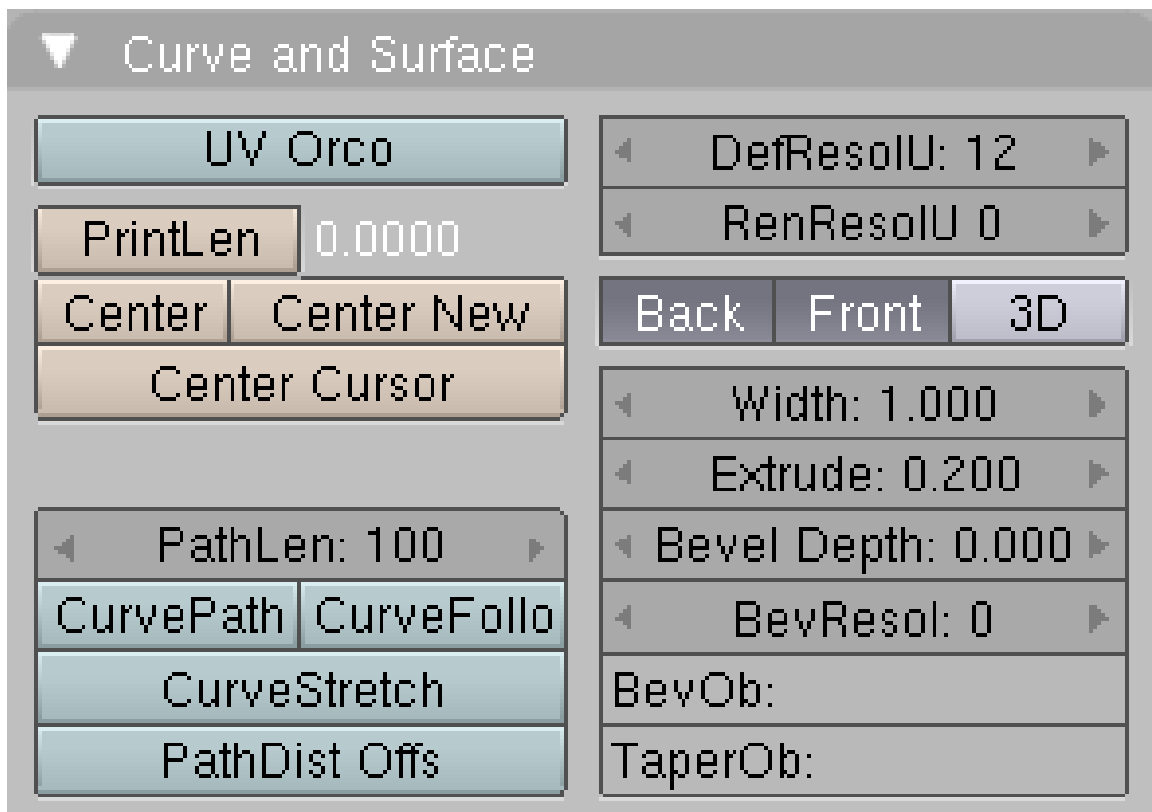


Figure 611

1. **Create a basic curve:** Press 'SPACE - Add - Curve - Bézier Curve' in case you have forgotten.

If you have been playing around with the control points on any curves so far, you will have noticed that they cannot be moved outside the flat plane made up by the X and Y axes. This is because it hasn't been enabled yet, so go ahead and enable it...

1. **Enabling 3D editing:** Locate and press the button labelled '3D' in the 'Curve and Surface' menu.

Your curve should now look like this in Edit Mode:

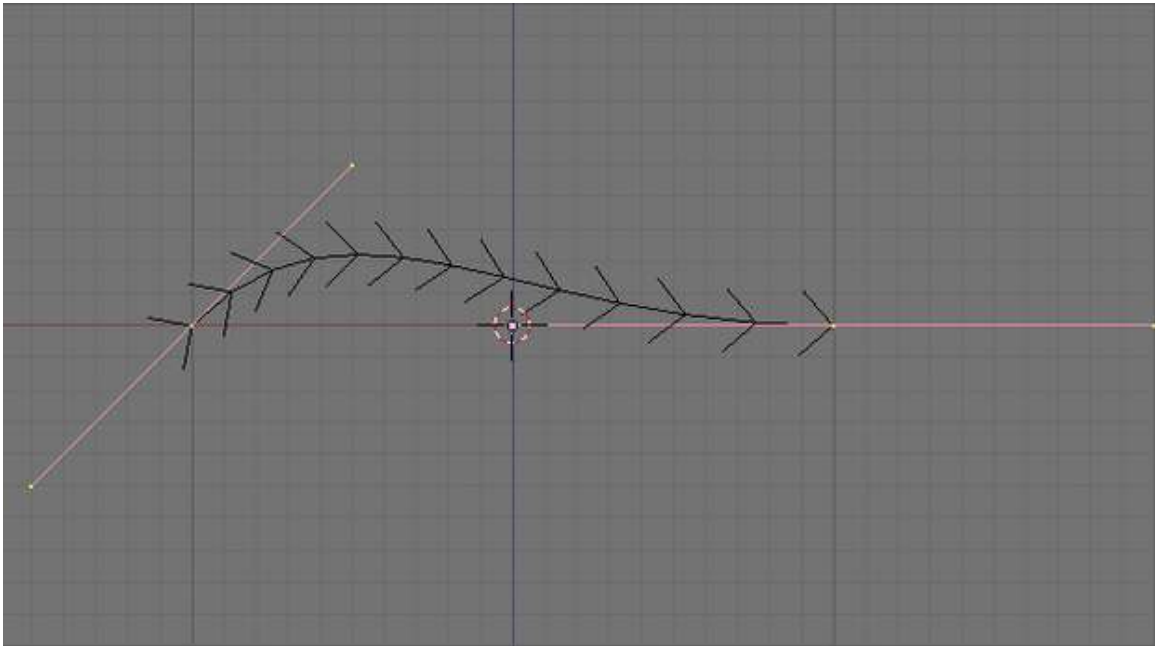


Figure 612

Our previously simple curve has now been covered with arrows! These indicate the direction of the curve, and also the normal size and tilt. But more of that later...

69.2. Getting started in 3D

Editing the curve should now be child's play, assuming you have already completed the tutorials on 3d meshes.

1. **Rotate the camera view:** Use the **MMB** or some Numpad keys to rotate the view so you can see the 3d axes.
2. **Move a point:** Select one of the control points (NB: A point, not a handle) and move it with **GKEY**. You may want to constrain its motion to the new Z axis using **ZKEY** twice.

If you rotate around the curve, you can see how the interpolation works in three dimensions, and that the arrows still follow its direction correctly. It is worth noting that they are all mostly vertically (for the moment at least), this will be discussed in the next section.

69.3. Extruding revisited

I'm sure you will be delighted to learn that we can still use the Extrude option on 3d curves! Try cranking up the value, and you will see a surface created through your curve, at right angles to the 'normal' arrows.

The Bevel settings will also work as before, but you probably don't need to use them unless you are editing a 'filled' curve (**CKEY**).

Also, The Subdivided option(W, 1) still applies while working with curves. Just select two(or more) handles of the curve that are right next to each other, and subdivide. You should have a newly created handle(s) in between the handles you selected to subdivide.

69.3.1. Bevel Objects

(Information on bevel objects to come...)

69.4. Curve Tilting and Normal-scaling

Use **TKEY** to tilt the curve, clockwise or counterclockwise (looking along the line of the curve).
Use **Alt-SKEY** to scale the curve, making it wider or narrower.

69.5. Convert Curve to Mesh

Use **Alt+C** [*ObjectMode>Menu>Object>Convert to>Mesh from...*].

Shape **typ 2D** = mesh with auto add faces and edges Shape **typ 3D** = mesh only edges and vertex

(To be expanded...)

70. Deforming Meshes using the Curve Modifier

70.1. Setting up your Mesh

While this example will be done with a cone primitive, you can use the default cube or another shape of your choosing so long as it can be loop subdivided along the axis you wish to curve. Delete the default cube if you're not going to use it as your base mesh, and add your chosen primitive. In this example, we'll be using a Cone.

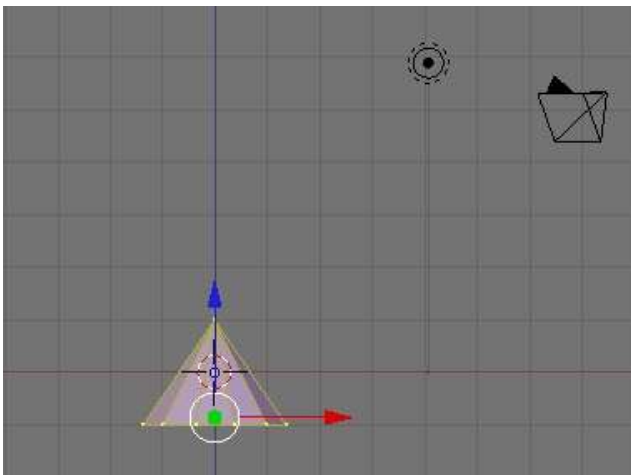


Figure 613 A basic cone in Edit Mode

- Set your view to the X-Y Axis (Top View) by pressing **NUM7**, the 7 on the keypad.
- Press **SPACE** > *Add* > *Mesh* > *Cone*
- A smaller number of vertices are needed, since this cone will become just the tip of the finished shape.
- Left Click and drag to reduce the number of vertices to 12, then click the Ok button. (Note: On newer versions of Blender you will have to hit **F6** to get to this option.)
- Press **TAB** to enter Edit Mode, or choose Edit Mode from the bottom of the 3D viewport.
- Switch views to the Z-X Axis by pressing **NUM1** (1 on the keypad). Your screen should now look something like the picture at left.
- If your cone is selected (one or more faces are pink, one or more vertices are yellow) press **AKEY** (Select/Deselect All) until they are deselected.
- Select the point of your Cone with the **RMB** and drag it upwards with the blue arrow. You can hold down the **CONTROL** key if you want to constrain the scaling to set units. For this example, 5 squares of height were added to the cone.

70.2. Extruding the Cone

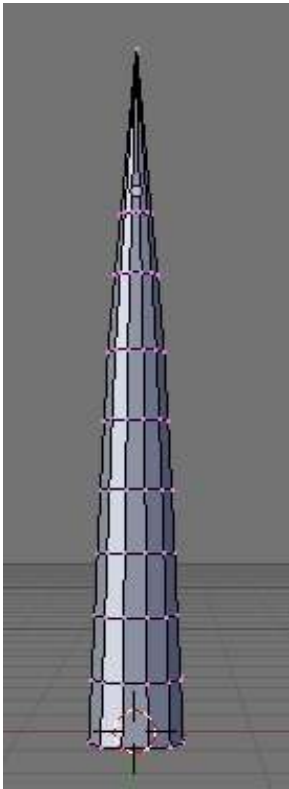


Figure 614 Extruded cone

- Still in Edit Mode, press the **NUM8** repeatedly to rotate your view to the underside of your cone. Select the center vertex of the circle, and hold down **CONTROL**, then press the **+KEY** on the keypad. This will select all of the adjacent vertices of the base. (You could also select the whole cone **AKEY** and deselect the top point by holding down **SHIFT** and selecting it with the **RMB** to achieve the same effect.)
- Go to front view by pressing **NUM1**
- Extrude the base of the cone by pressing the **EKEY** and choosing *Region*.
- Pull the new vertices away from the cone a distance and click the **LMB** to set them in place.
- Press the **SKEY** and widen the new base of the cone a bit.
- Continue extruding the cone, widening the base after each extrusion until you have at least 5 segments. The more segments you have, the smoother your curved cone will be.
- Another way of adding segments is with the Loop Subdivide tool. **CONTROL + RKEY** to bring up the purple placement ring, then **LMB** twice to add a subdivision.
- On Blender 2.6, the best way to do it is by selecting the side edges and subdivide them by pressing **WKEY** then clicking on *Subdivide*, after that you can select the number of cuts you want.
- You may find that your cone is quite large indeed now, so switch back to Object Mode (**TAB**) and resize it with the scale tool. While you're in Object Mode, try stretching the cone out a bit to make a nice long needle.

70.3. Making a Curve

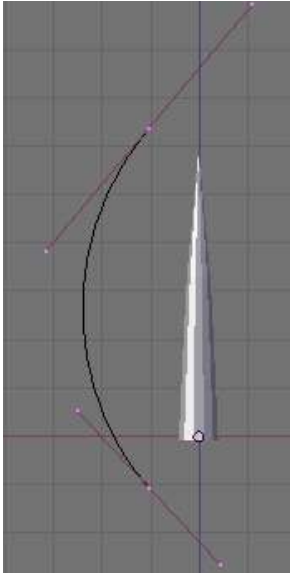


Figure 615

- In Object Mode, Z-X orientation (**NUM1**), make sure you have nothing selected and press **SPACE** > *Add* > *Curve* > *Bezier Curve*
- Rotate your curve 90° (holding down the **CONTROL** key to make it rotate in 5° intervals) Note- or just **R-KEY** and then type 90, so that it lines up with your cone. Move it off to one side so that its not hidden by your cone.
- Grab the center vertex of each end and adjust its rotation (**RKEY**) until you have a nice shallow dome shape.
- Scale the curve in size (**SKEY**) until it is larger than your cone.
- Look at the panel below the 3D space, you should see the Editing panel is selected

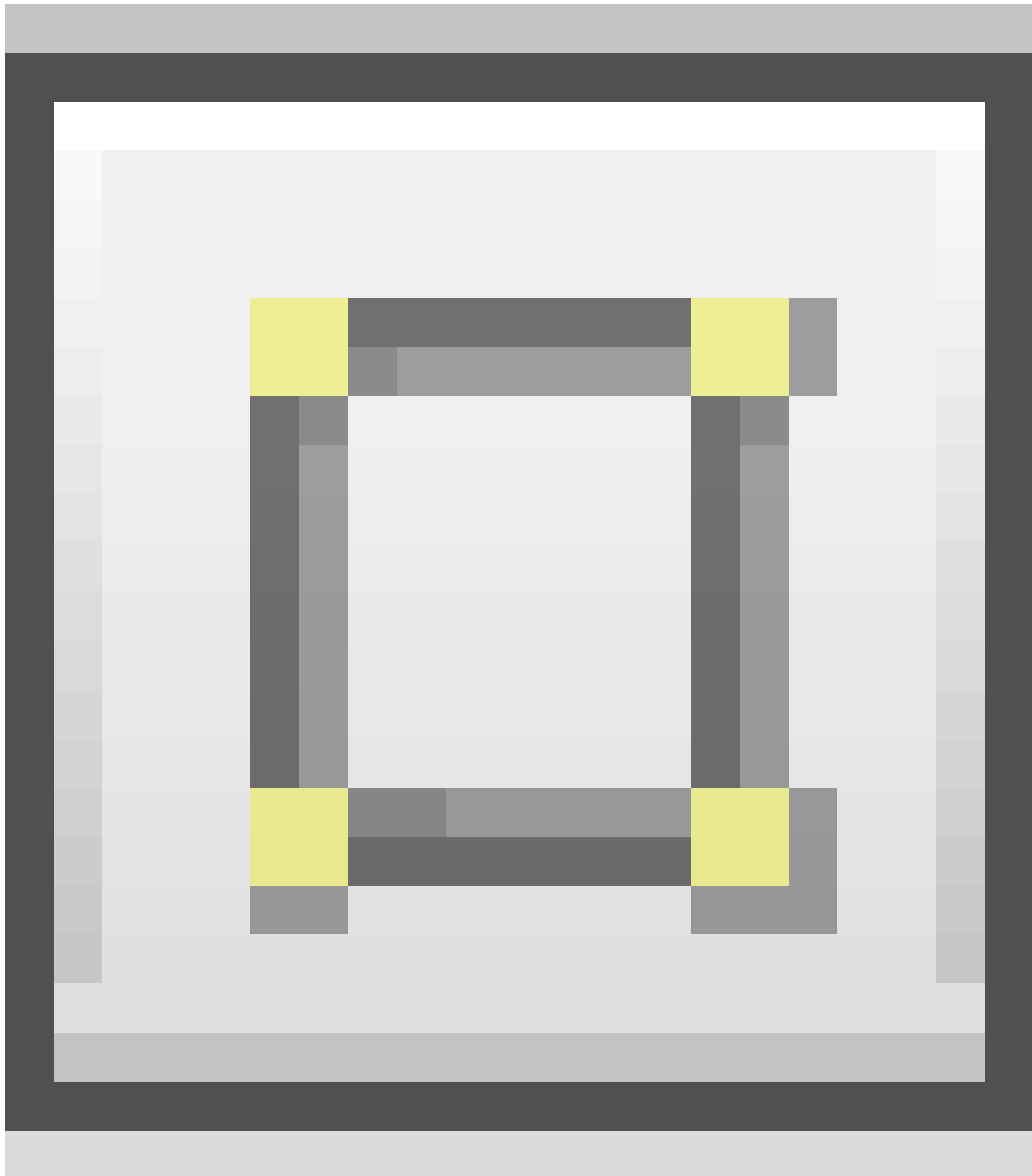


Figure 616

(F9KEY) Just below the panel buttons will be a window called Link and Materials. If your curve is selected, you will see a box with OB:Curve in it. This means the name of the curve is "Curve". This is important because that exact name is needed to deform the cone.

70.4. Applying the Curve to the Mesh

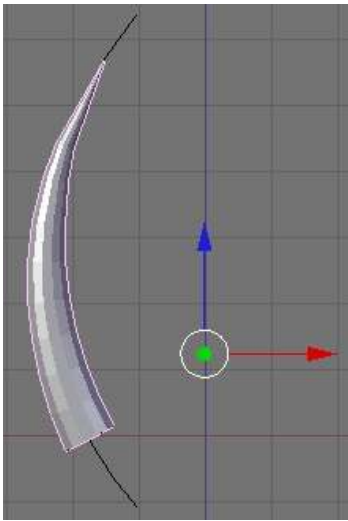


Figure 617 Modified cone

- Select the cone in Object Mode. In the Editing panel window (**F9KEY**), the rightmost panel has two tabs, Modifiers and Shapes. Choose the Modifiers tab if it is not already selected.
- Click the *Add Modifier* button and choose *Curve* from the popup list.
- In the OB: box, type the exact name of your Bézier Curve (it will be "Curve" without the quotations if you left it as the default.)
- Notice the six buttons underneath the OB: and VGroup: boxes (in the Modifier Panel). They are **X, Y, Z, -X, -Y, -Z**. They affect which plane the curve deforms. For this example, you'll need to select the **Y** button.
- Move your Cone so that it overlays the curve in the 3D space screen, and notice how it follows the Bézier Curve.
- You can modify the Bézier Curve as well as the cone repeatedly until you are happy with the design.
- If you switch to Edit Mode, notice that the cone returns to its straight orientation.
- To apply the deformation to the mesh permanently, in the Modifiers panel, click the *Apply* button next to your Curve modifier.

71. Making Your Creation Smoother

71.1. Getting started

To begin, open a new scene in Blender. Let's clean up the scene a little by selecting the lamp and camera in the scene, press M to move them to a different layer and click on the fifth layer from the left to place the objects. Go back to layer 1 and delete anything else. You may also want to get rid of the grid by opening the View Properties and turning off the Grid Floor and X & Y axes.

To block out our character, we're going to use an object type that is probably the least used and useful of any known to mankind. Let's hear it for... Metaballs!

71.2. Sculpting With "Lumps of Clay"

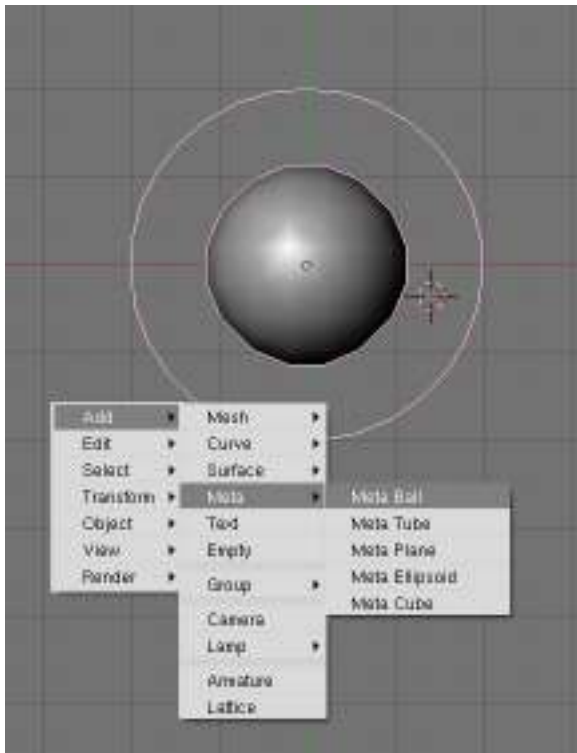


Figure 618

Go into the top view (important), press Space and add a **metaball**. Metaballs are a nifty, ancient piece of 3D technology that is useful for creating blobs. (Similar to lumps of clay, eh?) You create

simple primitive shapes and scale and rotate them to block out your character's shape. When the primitives come close to one another, they "bleed into one another" in much the same way that water droplets merge when they touch. Cool.

If you're using an earlier version of Blender that jumps out of Object mode into Edit mode when you create an object, then press Tab to switch back to Object mode, as you won't be able to scale your Meta primitive non-proportionally in Edit mode.

In Object mode, you can change these options for the entire Meta object, while tabbing into Edit mode gives you more options for the selected meta primitive, such as changing the type from Ball to Tube, Plane, Cube, ;etc.; (You can also make "Negative metaballs.")

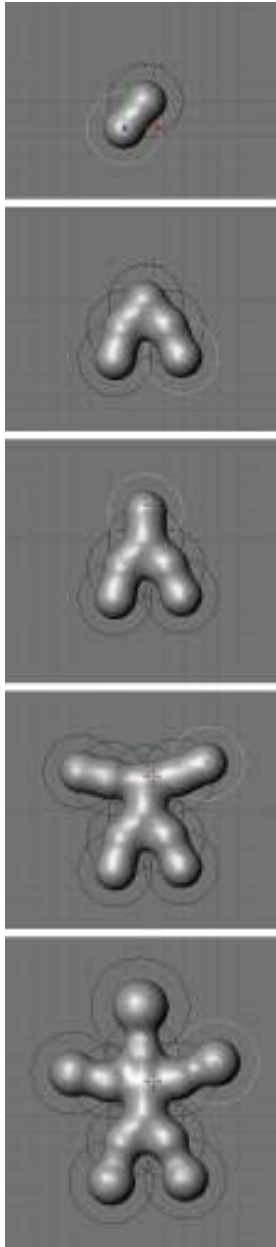


Figure 619

Press **Shift+D** to duplicate the Metaball, and place it where you like. Continue blocking out your character, building enough blobs to represent the limbs or forms you will need to sculpt your masterpiece.

The balls at the end of the limbs, were scaled **SKEY** larger, and then moved **GKEY** out a bit more.

Don't get carried away and put in too much detail at this stage: use as *few* shapes as you need. (This is supposed to be quick and fun, after all...)

71.3. Meta-mess!

You should still be in object mode.

Now that you've got something that resembles what you're after, select all the Metaballs and (be in object mode) type **ALT+C** -> "delete original", to convert it to polygons so you can actually do something with your blob.

Noob note: if you pick "keep original" you will still have the meta balls present, plus have a mesh version of the metaballs sharing exactly the same space. When you select "delete original", the meta balls are turned into the mesh, and the circles that the metaballs were originally in hang around, but are empty.

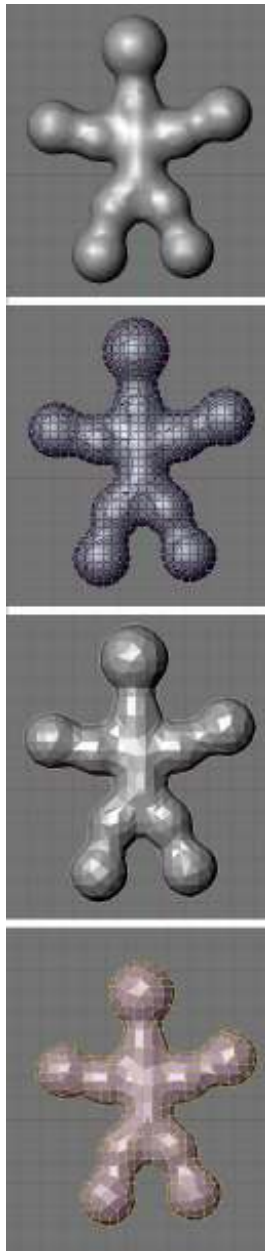


Figure 620

Still in object mode!

Delete any of those black rings left over from the metas and select your new polygon mesh.

If you Tab into Edit mode you will see terrifying ugliness instead of nicely gridded mesh. "Surely we can't be expected to create anything useful out of this!" you shriek. Take it easy, my friend. It's time to add a *Decimate* modifier (make sure you're in object mode when doing this to see changes).

Switch back to object mode!

The **Decimate** modifier (you will only see things change if you're in object mode) will do two things for us. Its primary job is to *reduce* the poly count of a mesh. A pleasant side-effect for our purposes

is that it will begin to *rearrange* the topology into a more manageable heap of triangles and quads. Keep reducing the Ratio slider below 0.5 until it becomes as coarse as you can stand. You want the lowest polygon base you can have that still maintains enough detail in the limbs and shapes you made with the Metaball phase.

One thing to watch for is that this process sometimes creates holes as it does its best to simplify the mesh. I find that usually you can slightly change the Ratio to fix the problem, but if you're still finding holes, check out the "Tips & Tricks" section at the end.

As you can see, this step greatly reduces the Face Count, which will be good later. Next we need to get rid of as many of those triangles as possible. Click Apply (in the modifier panel)

Switch to Edit mode!

Press A until all of the vertices are selected (turn yellow) and hit Alt-J to convert the faces from triangles to quads, which will subdivide better. Now you should have something you can work with.

71.4. Beginning to sculpt

Right next to the Modifiers panel is the **Multires** panel. (Note: In recent versions of Blender this is not its own panel--it is a Modifier.) A multires object has the options to add numerous levels of smooth subdivision to a mesh. While you can use the sculpt tools on any polygon or nurbs mesh, the great strength of sculpting with Multires is the ability to jump back and forth to different levels, quickly sketching out the gross form at lower levels, and adding finer detail at the higher levels. Add the Multires and Add a level; we're ready to sculpt!

Noob note: If nothing seems to be happening while you're trying to sculpt your mesh, it's because you haven't applied your Decimate. It must be locked down and applied before you're actually allowed to do anything to your mesh.

Noob note: Only apply multires once, doing it a few times will quickly increase the polygon count to amounts that will really slow blender down

In case you haven't noticed, Blender's different modes offer different tool sets and options. To make the sculpting tools available you need to be in **Sculpt** mode, accessible through the Mode drop-down in any 3D window header. You'll see two new tabs next to the multires panel, *Sculpt* and *Brush*. Sculpt has most of the options you'll need to begin shaping your mesh. Note that most of the different brushes in this panel have hot-keys which will save a lot of time (**G**=Grab, **D**=Draw, **S**=Smooth, *etc.*). Most important here is changing the brush size and intensity. Pressing F and dragging the mouse will resize the brush, while Shift-F will allow you to adjust the Brush Intensity. You can also turn on **Symmetry** to paint, for instance, both sides of a face at one time. This can speed up tasks tremendously, as long as your mesh is aligned to the axis properly.

72. Using Textures

This tutorial was created using Blender v2.49

Having said everything about Textures in Blender in the manual <http://wiki.blender.org/index.php/Doc:Manual/Textures>¹, many people find it still hard to grasp the concepts behind the texturing system and applying Textures.

There are some brilliant tutorials about texturing on the net, I personally like Texturing for Dummies² by Leigh Van Der Byl. It is free (as in free beer, not as in free speech like this book), so you should download and read it now, to take the most advantage out of this tutorial. We're going to see how to apply the texturing concepts within the Blender texturing system. I will explain it using an image texture and procedural textures, but be aware of the fact that the texturing system is quite complex, and more advanced maps are created with the use of a few combined textures.

If you're really into texturing you need to learn how to UV-Map³, there's no way around it.

72.1. Colormap



Figure 621 Image 1a: An image applied as a texture to a plane. Download texture^a

^a <http://en.wikibooks.org/wiki/%3AFile%3AWales-Tenby-Die%20Stadt.jpg>

¹ <http://wiki.blender.org/index.php/Doc:Manual/Textures>

² <http://www.leighvanderbyl.com/tutorials.html>

³ Chapter 80 on page 715

This is the most basic mapping type. You can use an image or a procedural texture to change the color of a material.

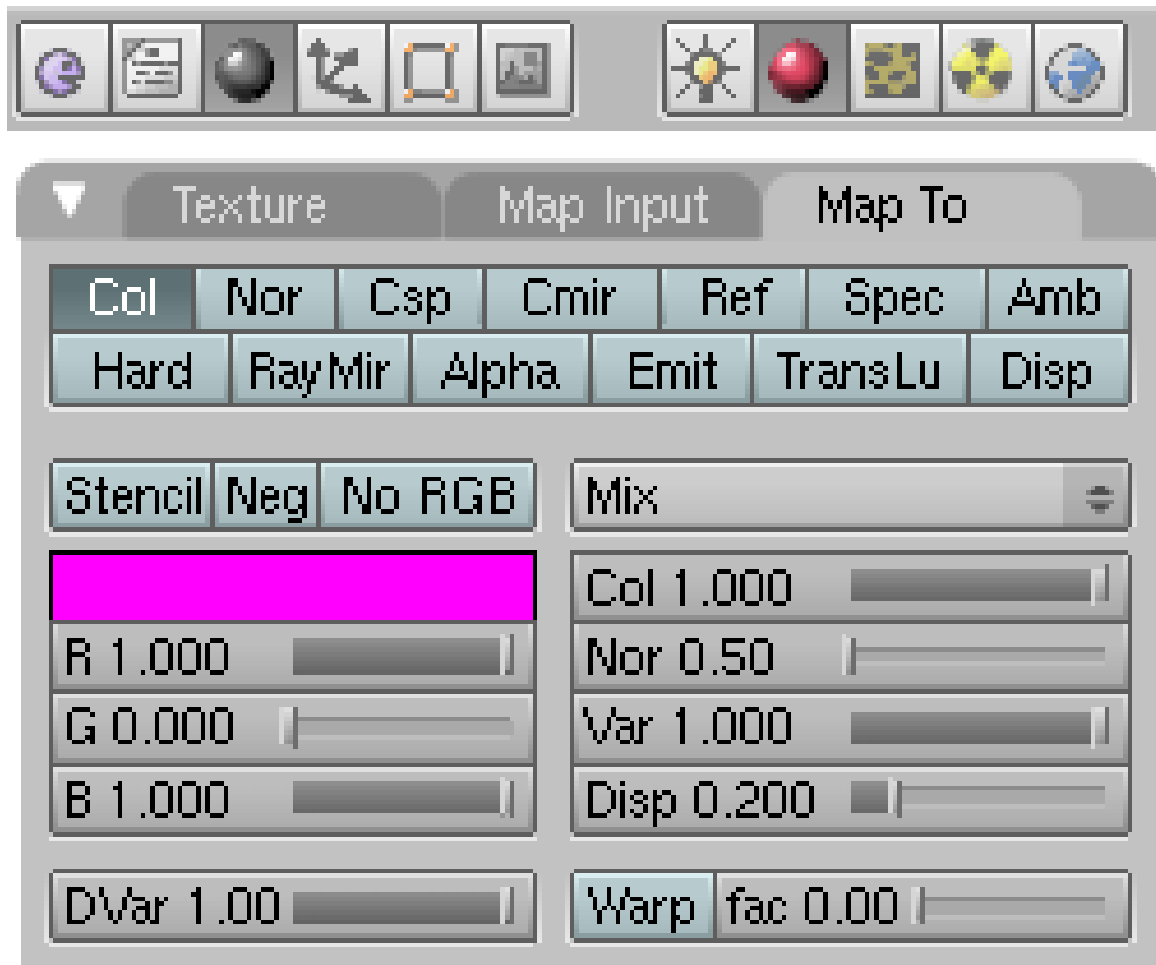


Figure 622 Image 1b: The *Map To* Panel for a colormap.

Using an image texture:

- Add an *Image* texture in the *Texture Buttons* and load the image (this is necessary also for UV mapped textures).
- The button *Col* in the *Map To* Panel uses the RGB information of the image to change the color of the material.
- The slide number button *Col* governs the blending amount of the texture, *Neg* inverts the colour values.
- If you use *No RGB* only the intensity information from the picture is used. The target color is then taken from the RGB sliders in the *Map To* Panel.

Using a procedural texture: The result depends on the type of value the texture provides.

- If the value is of type intensity (e.g. *Clouds*), the color is taken from the RGB sliders in the *Map To* Panel.
- If the value type is color (e.g. *Magic*), everything is handled as with image textures.

Note:

Color and intensity? What the heck?

An image texture (and other textures as well) may provide two different types of values:

1. RGB-Color information
 2. Alpha information. Now we have to take a look at a few different scenarios.
- If an image has no alpha channel and you
 - don't activate *Use Alpha* or *Calc Alpha* everything is easy - the texture provides only RGB-Color information. An intensity value is then calculated automatically if necessary.
 - activate *Use Alpha* the texture provides no intensity value and it is not calculated. If you now activate *No RGB* the intensity value is again calculated from the color information. So either you use *Use Alpha* and *No RGB* or none of the both.
 - activate *Calc Alpha* the texture provides automatically calculated intensity values. Now *No RGB* has no effect.

72.2. Diffusemap



Figure 623 Image 2a: Diffusemap settings with a small range (0.2).

A Diffusemap changes the amount of diffuse light the material reflects/absorbs. This is controlled by changing the reflectivity (*Ref*). A material with a rough surface may have the color white, but absorbs light to a greater amount than light with a smooth surface. For more on diffuse reflection physics, see Diffuse reflection⁴.

Noob note: Most of the settings described below will be found in the "Influence" section under the texture/material panel in blender 2.61

Let's test:

- In the Texture buttons (**F6**), in the Map Image panel, make sure *UseAlpha* is unchecked.
- In the Material buttons (**F5**), in the Map To panel, uncheck *Col* and check *Ref*.

⁴ <http://en.wikipedia.org/wiki/Diffuse%20reflection>

To understand the effect of this and all other mappings, we have to discuss the meaning of the NumButton *DVar* (Destination Value).

- White in a texture (or an intensity value of 1) will be mapped to the value of the *DVar* Button. So if you set *Dvar* to 1, white will be mapped to 1, if you set *DVar* to 0, white will be mapped to 0. The result is multiplied with the value of the *Var* number button.
- Black on the other hand (or an intensity value of 0) does not change anything, so the settings from the material will stay unchanged. The difference between the material settings and the *DVar* button is the range for the texture.

Let's take a look at an example.



Figure 624 Image 2b: The image used for the Colormap applied as a Diffusemap with a small range (0.2).

If we use the default settings for the Diffusemap the range of the values is small.

- *Ref* is 0.8
- *Dvar* is 1.0
- White pixels in the texture change the *Ref* value to 1.0, black pixels in the texture don't change the *Ref* value at all.

So the resulting **range** for *Ref* is $1.0 - 0.8 = 0.2$.



Figure 625 Image 2c: Diffusemap with full range (1.0).

When we change the *Ref* parameter in the *Shaders* panel to 0, we'll get a much larger range of *Ref* values from 0.0 (black pixels in the texture) to 1.0 (white pixels in the texture).

There are other ways to achieve a larger range, you could also set *Ref* to "Inverted" (click it twice, it will be painted in yellow), *DVar* to 0 and *Ref* in the *Shaders* panel to 1.

Note:

Why is this so complex?

Well, with complexity comes flexibility. You can use an image (or procedural texture) for a few mappings at once, e.g. Color, Diffuse, Alpha, Spec and the like.

72.3. Luminositymap

Luminosity is the property of self-illumination, i.e. of objects emitting light. So we change the *Emit* value with this texture.

This light does not illuminate other objects - you would have to use Global Illumination (e.g. Yafaray) to use an emitting object as a true light source. Radiosity rendering does not work with Luminositymaps.

Everything that was said about the *DVar* value applies here exactly as for Diffusemaps.

72.4. Specularitymap

Specularity is the second most important material attribute. Specularity fakes the reflections of light sources. Of course you can modify the general specularity with a texture. In Blender you can change three different attributes related to specularity:

1. *Spec*: The degree of specularity. You can't set values above 1.0 with Specularitymaps, but you could invert the texture and set *Spec* on the *Shaders* panel to a higher value and *Dvar* to the

lowest value of your range. So the texture actually lowers the specularity in all areas, that shall have a low specularity.

2. *Hard*: The hardness of the specular reflections. Sometimes called "Glossiness". A *DVar* of 1 is equivalent to a hardness of 130. Use the same method as described above to achieve a greater range for the hardness.
3. *Csp*: The specularity color.

You will probably use some kind of "Dirtmapping" (stenciling) to change the specularity in certain regions of an object, like an often touched object, that is shinier on the used or touched parts.

For more on specular reflection physics see [Specular reflection](#)⁵.

72.5. Reflectionmap

A Reflectionmap would be used to fake real raytracing reflections, either because you don't want to use raytracing, or you don't want to create a scene, or you need a special effect you don't get with raytraced reflections. A Reflectionmap would be a Colormap, typically an Environment Map⁶. Since you can use prerendered Environment Maps, you can fake a surrounding for your object. Sometimes it is sufficient to use a simple Colormap that just bears the right colours, e.g. if you want to create a stormy sea it is not necessary that the clouds above are truly reflected in the water.

Environment Maps are a bit more complicated to create, see the respective section in the manual.

A very different kind of texture would be used to change the amount of raytracing reflections, the *RayMir* value. This is useful for something like a stained or dirty mirror. Simply click the *RayMir* button in the *Map To* panel and lower the *DVar*.

72.6. Transparencymap

Transparency- or Alphamaps change the (partial) visibility of an object. They don't have to be transparent themselves (though it doesn't hurt, see [Make a material partially transparent](#)⁷). You will often use an image as Colormap and Transparencymap together. A nice example for a Transparencymap is in the paragraph *Translucencymap*.

There's one catch though, you can't change Fresnel transparency with a Transparencymap. So you have to use plain transparency, or fake the Fresnel effect with the method described in the section [Map Input](#)⁸.

5 http://en.wikipedia.org/wiki/Specular_reflection

6 http://wiki.blender.org/index.php/Doc:Manual/Textures/Types/Environment_Maps

7 [Chapter 73 on page 669](#)

8 http://wiki.blender.org/index.php/Doc:Tutorials/Textures/Map_Input_Techniques

72.7. Refractionmap

Well, there's no such thing as a Refractionmap. You can't change the IOR with a texture. You may try and use Environment Maps to fake refraction, but the result is often not worth the effort (for stills, in an animation it might not be that noticeable).

72.8. Translucencymap

Translucency is a material property of all semitransparent materials, like frosted glass, paper, plastic, cloth, skin, stained glass and the like. It allows objects to be lit from behind.

Totally clear glass does not show where the lightrays travel through it. But if the glass is dirty or uneven you will see the path of the light. In the first example (*Translucency 1*) the glass material has a transparency of about 0.5 and a translucency of 0.78. So the glass is brighter where the lightrays hit the surface from behind. Additionally a Transparencymap - which sets Alpha to 1 - lets the surface appear to be stained. If we use the same map also as a Translucencymap (*Translucency 2*), the stain appears to be lit - which it would be of course also in reality.



Figure 626 *Translucency 1*: Material with Translucency and a Transparencymap, but without a Translucencymap. This is an example from the releasenotes of Blender v2.32.



Figure 627 *Translucency 2*: Material with Translucency and a Transparencymap and a Translucencymap.



Figure 628 Material settings for example *Translucency 2* (blue glas).

The settings for the material (*Z-Transp*, but raytracing shadows):

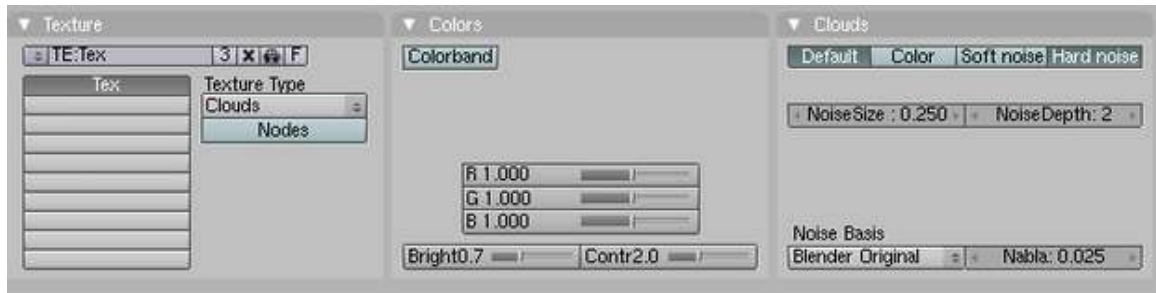


Figure 629 Adjusted *Clouds* texture.

The texture itself is a *Hard Noise Clouds* texture, with modified *Brightness* and *Contrast*.

Note:

To get the shadow and those colors as in the example images set the light to use *Ray Shadow* and the plane to receive transparent shadows by enabling *TraShadow* in the shaders tab in the *Material* buttons.

72.9. Bumpmap/Normalmap

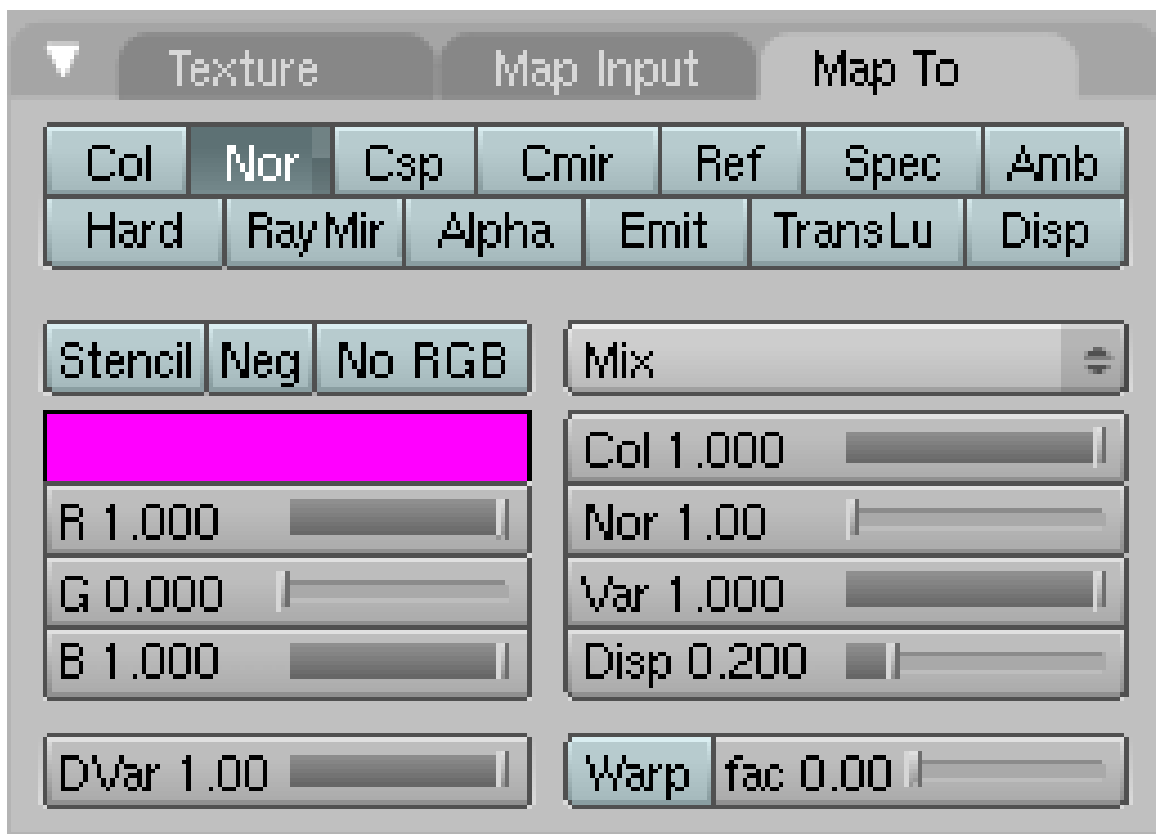


Figure 630 Settings for a Bumpmap in the *Map To* panel.

Bumpmaps⁹ are a technique to create the illusion of geometry. They could be used for something like a canvas or any small structures that are either difficult to model, or are too computationally expensive. As long as you don't get too close with the camera you won't notice the difference.

They come in several different flavors:

- A Bumpmap texture in the nearer sense is a greyscale image or procedural texture mapped to *Nor*. Don't activate the button *Normal Map* in the *Texture Buttons* for a Bumpmap.
- A Normalmap is an RGB image whose color information bears the information about the direction of the surface normal, also mapped to *Nor*. For this you need to activate the button *Normal Map* in the *Texture Buttons*.

See also the manual about the differences between Bump and Normal Maps¹⁰.



Figure 631 The bumped image.



Figure 632 The Bumpmap, a simple greyscale image.

In the example the bump map was used additionally to the colormap. The *Nor* slider in the *Map To* panel sets the depth of the bumping.

⁹ <http://en.wikipedia.org/wiki/Bump%20mapping>

¹⁰ http://wiki.blender.org/index.php/Doc:Manual/Textures/Maps/Bump_and_Normal_Maps

72.10. Displacement Map

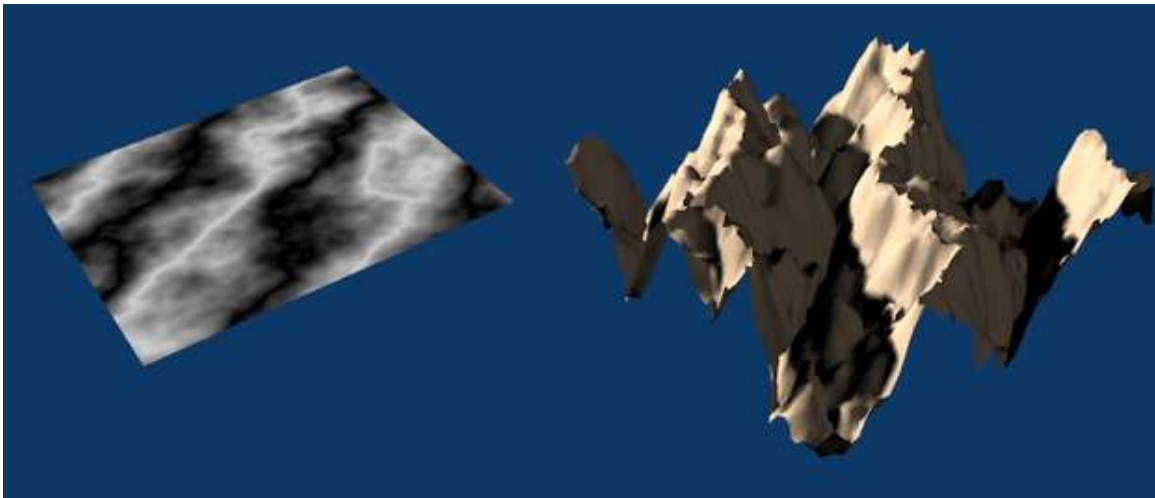


Figure 633 A displacement map deforms the mesh

A Displacement Map creates real geometry, i.e. it moves the vertices of a mesh. Therefore you need a lot of vertices. You can use it as a texture mapped to *Disp*, or even more flexible using the Displacement Modifier¹¹.

72.11. Dirtmap



Figure 634 Our picture has seen some bad times.

¹¹ <http://wiki.blender.org/index.php/Doc:Manual/Modifiers/Mesh/Displace>

I will try to explain what a Dirtmap is and how you apply one. There are different things that people mean when they talk about dirtmapping.

- Sometimes a Diffusemap is meant.
- Dirtshading is often used as a term for Ambient Occlusion.

With a Dirtmap I simply mean any texture that breaks the clean, uniform, untouched look of image or procedural textures. You may do this by simply combining some partially transparent textures resp. textures that affect only parts of the picture in a semirandom way. You can also select parts of the object with a stencil¹² texture. And of course you can (and will) combine both of these methods.

As long as two textures don't affect the same property you can simply apply one after the other. To select only parts of your texture to apply another texture with the same properties use stenciling. Or use stenciling as a convenient method to separate two different sections on your texture.

You need at least three textures for this.

1. The basic texture (e.g. color).
2. The selecting (stenciling) texture.
3. The Dirtmap affecting the same or other properties then the first texture.

The stenciling texture has to provide an intensity value, so if you want to use an image (or other RGB) texture you must activate *No RGB*.



Figure 635 The stenciling texture.

The image texture used as a dirtmap for our example contains some semirandom noise.

¹² <http://www.cgcookie.com/articles/2008/12/21/blender-intro-to-stencil-maps>

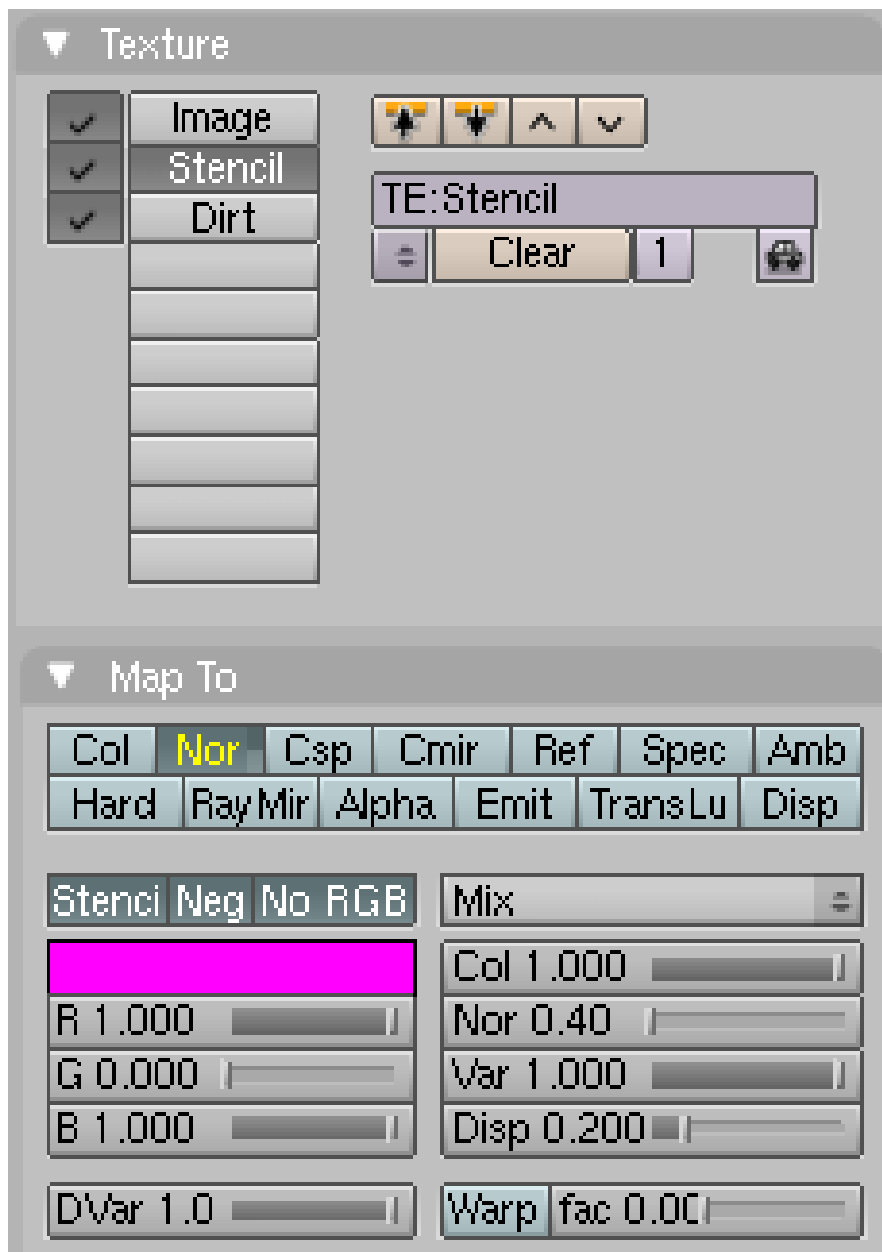


Figure 636 Settings for the stenciling texture.

Nor for the stenciling texture was not really necessary, but it adds a nice 3D effect. Since the stenciling texture is mostly white, I had to use *Neg* to use the black pixels as mask.

The third texture ("Dirt") affects *Col*, *Nor* and *Spec*.

72.12. Use UV Coordinates for your Maps

It is often necessary to create UV coordinates for more complicated textures/objects. UV coordinates only describe the geometry of the mapping, i.e. which pixels to map on what face. You can use the UV coordinates to create any mapping you want.

1. Create an UV map for your object. This is beyond the scope of *this* tutorial, see UV Map Basics¹³ and the section about UV Mapping in the manual.
2. Don't turn *TexFace* in the *Material Panel* in the *Material buttons* on. Though *TexFace* is a very simple way to use UV textures, the normal texture mapping methods are much more flexible.
3. Load the image you used for UV-mapping (or any image with the same size) as an image texture. Change *Map Input* to *UV*.
4. Now you can use this image - or multiple different images - for your maps, including color-mapping.

The advantage of this technique is that you have much better control over your texture. Even if you just want to texture something like a cube shaped object, you're often better off with UV mapped textures.

73. Using a texture to make a material partially transparent

This tutorial was created using Blender v2.49

To understand the collaboration of textures and material we will use the opacity of a material - the alpha value - as an example. We will show how to set the opacity/transparency of a material with different types of textures.

The Alpha value of the material is set with the *Alpha* slider in the Material panel. This is the basic Alpha value. If *Alpha* is 1, the material is fully opaque, if *Alpha* is 0, the material is fully transparent.

Now you change that basic Alpha value with a texture.

1. If the basic Alpha value is 1 you can only lower it with a texture, because 1 is the maximum value anyway.
2. If the basic Alpha value is 0 you can only increase it with a texture.

The target value of Alpha that can be achieved with the texture is set with the **DVar** button in the *Map To* panel. If *Alpha* is 0 and *DVar* is 0 than nothing will happen. You can set either:

1. *Alpha* to 1 and *DVar* to 0, then the texture will lower the Alpha value, or you can set
2. *Alpha* to 0 and *DVar* to 1, then the texture will increase the Alpha value.

Both methods are possible and equal in the result, but for clarity we will always use method number 1. There are more combinations possible, so it may be that you find tutorials with different settings.

73.1. Using a greyscale texture - white for opaque



Figure 637 Image 1a: Greyscale Image of leaf

In **Img. 1a** the black areas shall become transparent, the white areas shall become opaque.

- Load the image as *Image* texture, turn off *Use Alpha*.
- Set the *Alpha* value in the material buttons to 1. This is the default value anyway.
- Activate *ZTransp* (or *RayTransp* if you need refraction)
- The *Map To* panel:
 - Turn off *Col* (or else the texture would also affect the color of the material)
 - Turn on *Alpha* inverse, you have to click it twice. We need inverse here, because the texture shall affect the material from its black parts.
 - Turn on *Spec* inverse (you don't want your transparent sections to show specular highlights)
 - Set the *DVar* value to 0. The default value is 1.



Figure 638 Image 1b: Material settings for the texture

Now let's take a look at the texture.

- Where the texture has a value of 0 (e.g. black) it will affect the Map To values fully (because we have used inverse).
- Where the texture has a value of 1 (e.g. white) it will do nothing.
- Where the texture has a value in between 0 and 1 it will affect the Map To values partially.



Figure 639 Image 1c: Result of applying the Greyscale Image to a plane

The result is shown in **Img. 1c**, the texture applied to a plane rendered against a white world background.

If your texture is inverted, i.e. black shall become opaque and white transparent, simply use the normal *Alpha* and *Spec* setting in the *Map To* panel.

That's it. It's basically the same for all settings, the only problem that remains is sometimes to know which value a texture will give if you use transparent textures or colored textures.

73.2. Using a partially transparent texture to make the material transparent



Figure 640 Image 2a: Using a partially transparent texture

Often we have a partially transparent texture, and want to directly use that transparency of the texture to make the material transparent too. We have now to think a bit about the values that are generated by the texture itself: If we load the image texture with *Use Alpha*, the Alpha value of the image is used as input value for *Alpha* and *Spec* in the *Map To* panel. So a fully transparent pixel in the image has an Alpha value of 0, a fully opaque pixel has an Alpha value of 1. This is exactly the same situation as above, simply turning on *Use Alpha* and *Premul*(tiplify Alpha in Advance) will give use the same result as with the greyscale image.

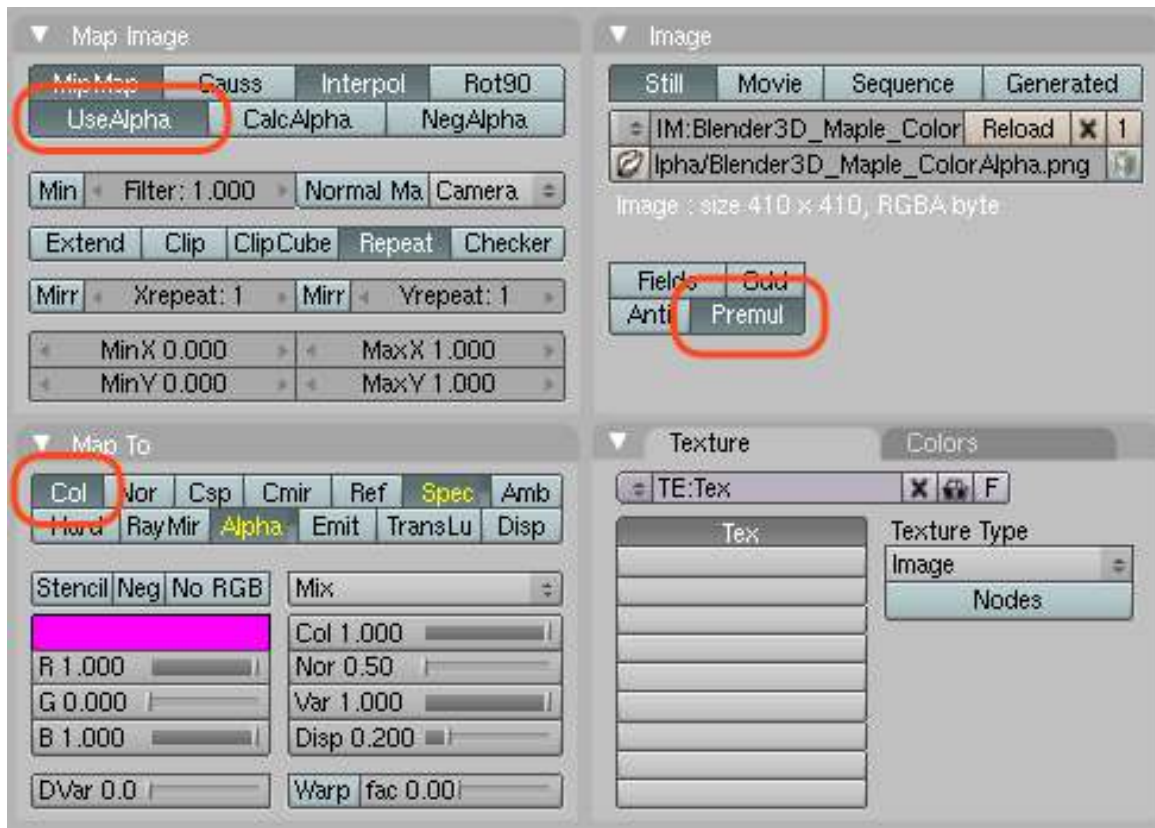


Figure 641 Image 2b: Texture settings for a RGB/Alpha texture

So:

- Turn on *Use Alpha* in the *Map Image* panel of the texture buttons.
- Turn on *Premul* in the *Image* panel. This is necessary to get real partial transparent borders.
- Turn on *Col* in the *Map To* panel additionally.



Figure 642 Image 2c: The result

The rendering result is pretty similar to **Img. 2a**, only that it is affected by lighting (and casts shadows).

Don't forget to turn on *TraShadow* in the *Shaders* panel for the object that is **receiving the shadow**.

(Noob Note: *In this case add a plane colored white just below the "leaf." This is the object the you should turn on **TraShadow** with as this is the object that will receive the shadow. This is also providing that you have a light source properly positioned.)*

74. Creating Basic Seawater

[ed. note: Need a much more basic introduction to what materials, textures, maps, and all the accompanying terms are with illustrative examples before diving into a specific sea-water example. Much more effective learning when you know what you're changing.]

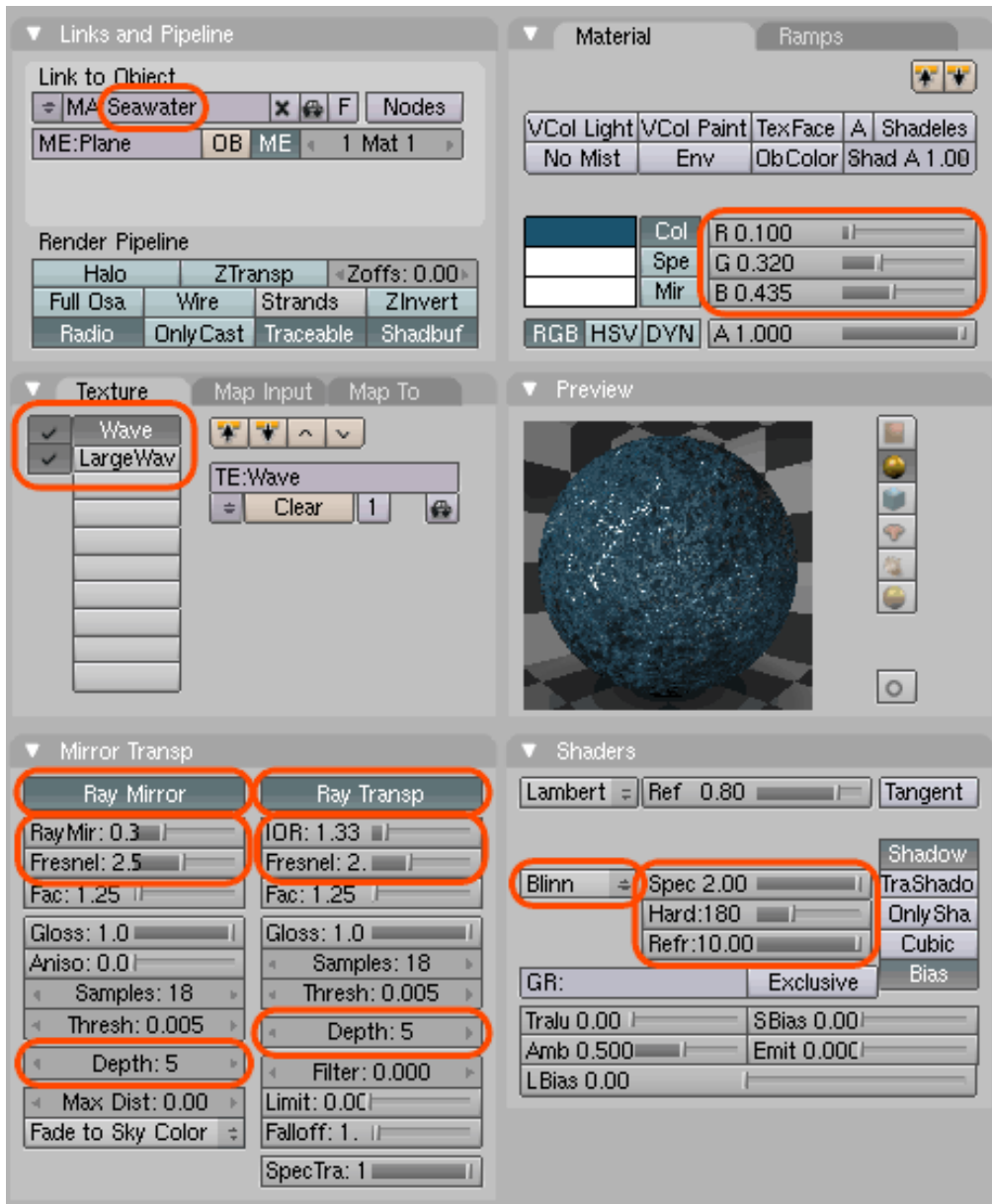


Figure 643 Settings for the seawater material

75% of the Earth's surface is covered with water. In homage to this great fact, we will develop your materials skills first by creating basic seawater.

74.1. Create 3 linked planes

First we create a new file in Blender and delete the default cube by pressing **XKEY** and confirming the popup dialog. Now switch to top view with **NUM7** and enter **SPACE > Add > Mesh > Plane** to create a plane. Then scale it up to 20 its original size with the **SKEY** the way you've already learned in one of the earlier tutorials. Go to the side view with **NUM3** and duplicate this plane *two more times* using **Alt-D** (not Shift-D), moving the plane down on the Z axis by two grid spaces each duplicate. This will make the transparency of the water more realistic once we set it. Using Alt-D rather than Shift-D makes a linked duplicate, so that the changes we make to one plane effect the other two.

74.2. Create material

Now off to the actual texturing work. Select any one of the planes and press **F5** to bring up the Material Buttons in the Buttons Window. You will probably find two new small windows appearing here: one called Links and Pipelines and the other one Preview.

Click the 'Add New' button in the 'Links and Pipelines' tab to create a new material named 'Material.001' or so. To make life easier we'll rename it to something meaningful like 'Seawater' by simply clicking it and typing in the letters, as shown here (SHIFT+DELETE in field to clear):

Now, on the same tab, give the seawater material a color of RGB (0.100, 0.310, 0.435). Find the tab that reads 'Mirror Transp' and click it. Click on 'Ray Mirror' and 'Ray Transp'. For the "Ray Mirror" box, move the 'RayMir' slider to 0.3, the 'Fresnel' slider to 2.5 and the Depth to 5. For the 'Ray Transp' box, move the 'IOR' slider to 1.33, the 'Fresnel' slider to 2.0 and the Depth to 5. This will give the water realistic transparency and reflection. Also click the 'Shaders' tab, change 'CookTorr' to 'Blinn', move 'Spec' to 2.000, 'Hard' to 180 and 'Refr' to 10.000. This will make the water look more glossy.

Now we'll add a procedural texture to our seawater, which will give it a "wavy" look. Click the Texture button (looks like bricks) or press **F6** to view the texture buttons subcontext. Click on the knob to the left of the texture name and select the "Add New" button. This creates a new texture named "Tex.001" or so. Click on the name and change it to "Waves".

Go to the Texture Type pull-down (**F6**) and select 'Clouds'. On the Clouds tab change 'NoiseSize' to 0.050. Our Waves texture is ready; next, we will refine how it is applied to our Seawater material.

Noise Size increases the size of the noise, in this case, the clouds. Soft Noise blends the intensities and reduces the contrast. Makes a mellow effect, like soft waves. Hard noise creates a high contrast, and brings out individual 'shapes'.

If you want to add more detail to your water, add another texture and rename it to "LargeWaves". Make it a cloud texture like the previous one, but make it's 'NoiseSize' 0.300 and use 'Hard noise'.

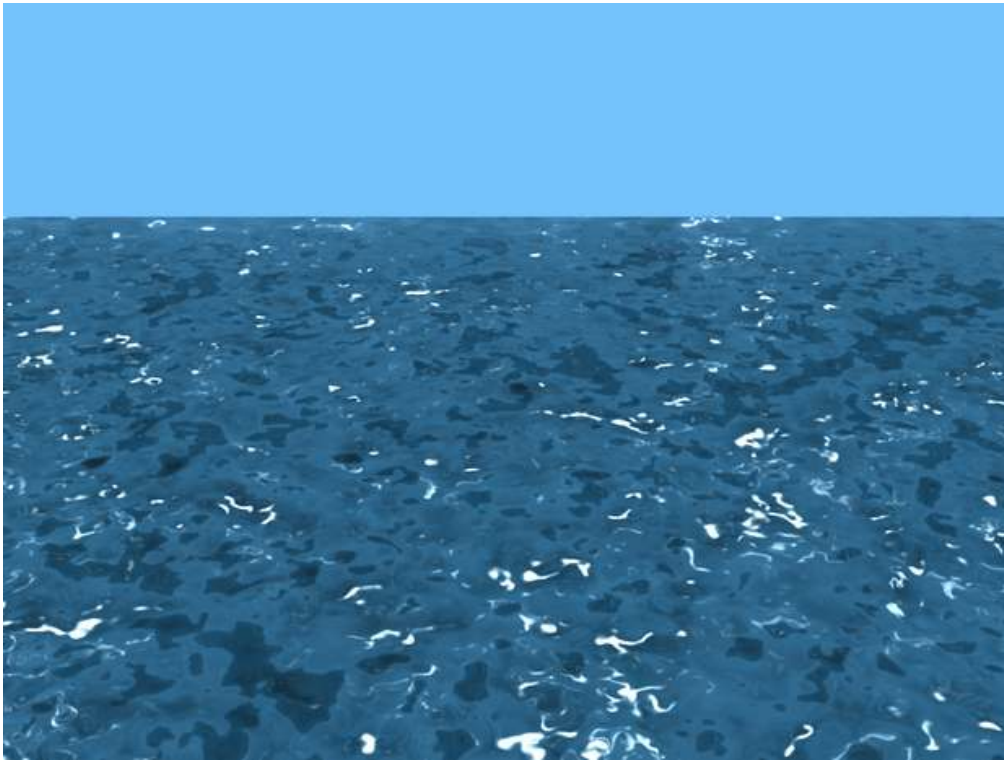


Figure 644 The final rendered image

Left click on the Materials button (looks like a red sphere) to return to the material buttons subcontext. Look at the Texture panel, and you'll see that the "Waves" texture has been automatically associated with the Seawater material.

Select the 'Map To' tab. Click the 'Nor' and 'Spec' buttons so they're selected and have white text (the white text indicates a positive mapping). Click the 'Hard' button twice so it's selected and has yellow text (the yellow text indicates a negative mapping). Click the 'Col' button so it is not selected, this button will show any color in the texture which we do not want. Find the 'Nor' Slider and move its value to about 5.00.

If you created the "LargeWaves" texture, select the "LargeWaves" texture under 'Texture and Input', go to the 'MapTo' tab, deselect 'Col', select 'Nor' and move the 'Nor' slider to 7.00. Do not select 'Hard' or 'Spec' this time.

For lighting press **Space > Add > Lamp > Sun**. You shouldn't need to move the sun or change any of its settings. Finally move the camera to the edge of the plain and move it up towards the sky a bit.

Go to the Scene tab (**F10**), and look for the six buttons next to the big render button. Deselect all these, leaving only the 'Ray' button selected. This will tell Blender not to render some features in our scene that we really don't need. Go and press **F12** to render the water, it may take a while depending on your system.

Admire your water, and maybe drink a tall glass of something refreshing!

74.3. Extra Practice

This tutorial might also help you make even more realistic water: [Link¹](#)

¹ <http://forum.nystic.com/viewtopic.php?f=34&t=1445>

75. Mountains Out Of Molehills 2

This tutorial shows you how to use displacement mapping to make a simple environment.

1. Make a grid. (Add/Mesh/Grid) 32x32 will do just fine. Be sure to create a Grid instead of a Plane, or you'll end up with just a flat plane! This is because a Plane only has four vertices to manipulate - one for each corner - while a Grid has many, many more. A 32x32 Grid, for example, would have over 1000 vertices. The more vertices there are to manipulate, the more effect displacement mapping will have on the finished product.
2. Set it smooth. (Editing/Link and Materials/Set Smooth)
3. Make a new material for it. (Shading/Material/Add New)
4. Make a new texture for the material. (Shading/Texture/Add New)
5. Go to Shading/Texture Buttons. You can see your newly created texture there now.
6. Change Texture Type to Clouds.
7. Change the name of the texture to be more descriptive. For example GroundDisp or something similar.
8. Go back to Shading/Material buttons. You can see our cloud texture applied now, but it's not applied correctly yet. Let's fix this next.
9. Go to Shading/Map To. This defines how the selected texture is mapped on our material. Uncheck Col, check Disp on, and set the Disp slider to a value like 0.200
10. Set camera and a few lights to the scene. (This is already done as part of the default scene in recent versions of Blender, such as 2.49)
11. Render.

Several individuals, when working through this tutorial, had trouble getting anything more than a flat plane. A few solutions were proposed (most are still visible on the Talk page), and while each worked in its own way, the vast majority of the time the problem stemmed from creating a Plane instead of a Grid. Be sure to follow each step very carefully to ensure you don't miss anything or do a step incorrectly.

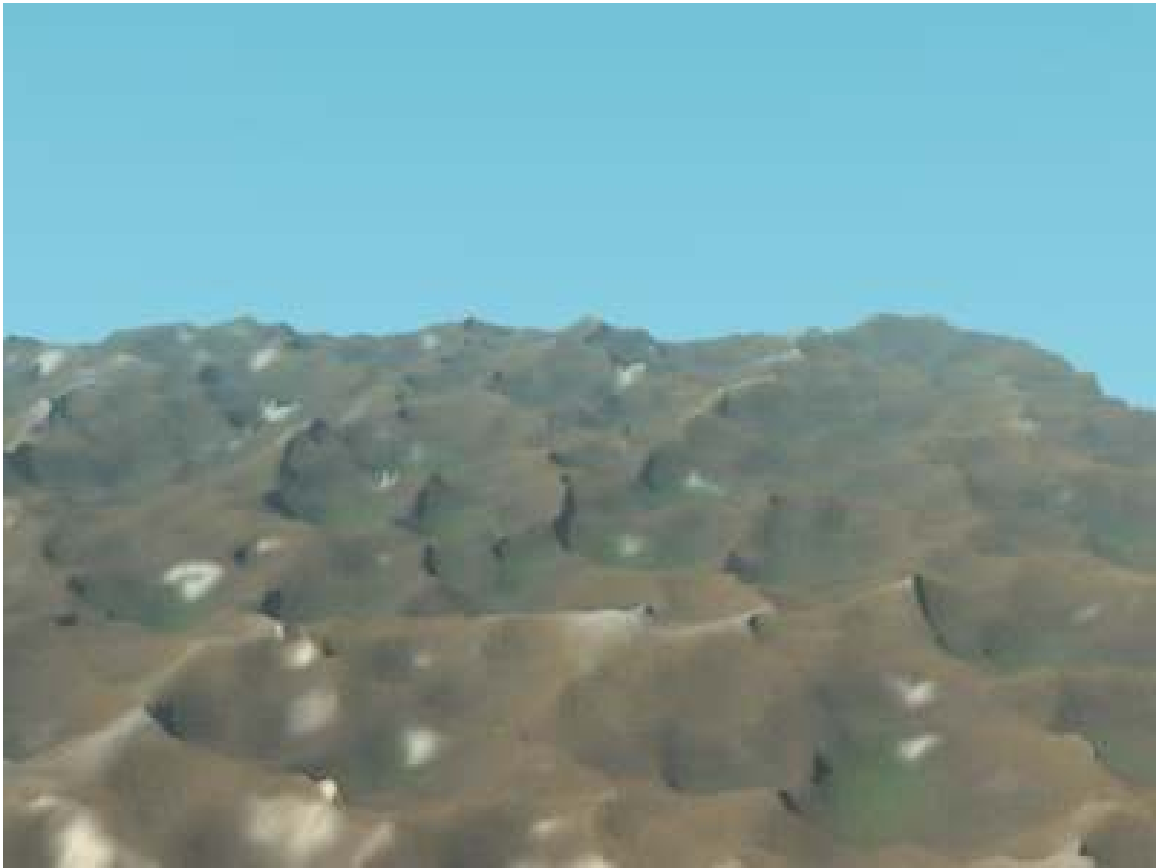


Figure 645

You can tweak the environment easily by changing Nor value in the Shading/Map To. This defines how strongly the displacement texture affects the material.

You could also add subsurfing to the ground area to get smoother results. Also feel free to tweak the texture and try out different alternatives.

Once your mountain looks good, try adding some Mist.

1. Select the "Mist" button on the "Mist/Stars/Physics" Tab among the World buttons
2. Add a Cloud Texture to the World and make it blend from white to gray

The end result is something like this:

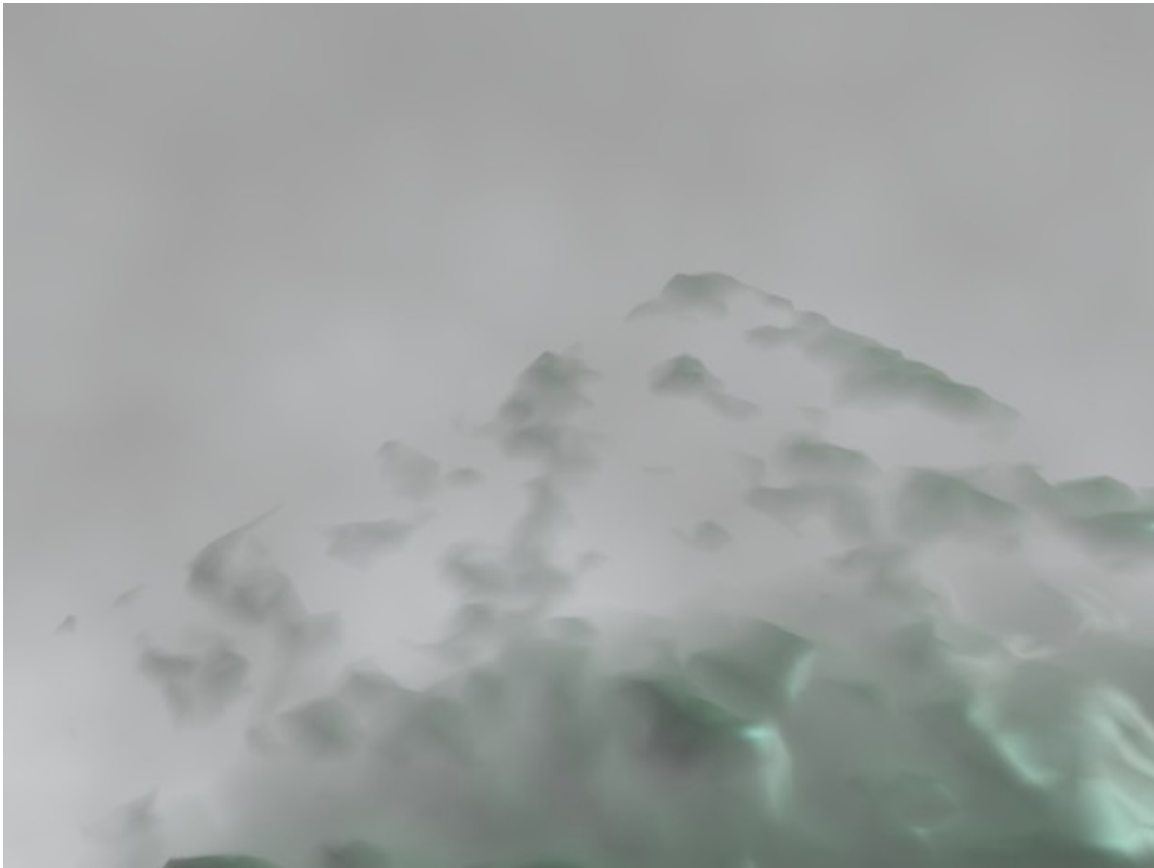


Figure 646

76. Basic Carpet Texture

76.1. Goal

I am using a basic scene that I quickly set up before I started to create the carpet material. It shows a monkey (suzanne), a plane, camera, 3 area lamps and 1 spot lamp.



Figure 647 The basic scene setup

The purpose of this tutorial, is to highlight the power of blenders built in shaders and procedural textures to create a carpet material to use in your scenes.

NOTE:For those of you needing help getting a similar scene to the one above, here are some axis positions,etc. to help out:
(Spot-Lamp)-X=1.62,Y=0.86,Z=6.74;rotation-X=37.26,Y=3.16,Z=181.34;
(Area-Lamp-1)X=4,Y=3.27,Z=4.12;rotation-X=54.67,Y=-18.59,Z=-109.47;
(Area-Lamp-2)X=-2.07,Y=-2.08,Z=4.85;rotation-X=29.37,Y=-28.98,Z=355;
(Area-Lamp-3)X=0.315,Y=-2.89,Z=4.29;rotation-X=49.23,Y=-10.63,Z=6.68;
(Monkey)X=0.0
5,Y=0,Z=0.42;rotation-X=58.61,Y=-16.07,Z=23.245;DIM(dimensions)X=2.734,Y=1.969,Z=1.703;
(Plane)x=0,y=0,z=0;(no rotation);Dimensions: X&Y=14.30,z=0.

Alternatively, you could also download a pre-made file¹ in case that you're in a hurry.

NewbyNote: With these settings, I get a big white-out.

¹ <http://www.mediafire.com/file/wwkzd9be907681g/basic-scene.blend>

NoobNote: Try adjusting each of the area lamps Dist(ance) value. This helped me.

NoobQuestion: How does the above coordinates help me at all? I haven't found anything that tells me were in 3D space object actually is or the objects dimensions little alone control them. Nothing in the Tutorial up to now even gives a clue to this.

NoobAnswer: Hit 'N' to bring up the Transform Properties window and enter the values there.

76.2. The Basic Material and Shader Settings

For the basic material for the carpet set the color settings of your material as follows -

Col (R 0.714) (G 0.134) (B .134) Dark Red

Spe (R 0.590) (G 0.210) (B 0.084) Reddish Brown

Mir (R 1.000) (G 1.000) (B 1.000) White

Change the specular shader to ('phong') and leave the default lambert diffuse shader as it is. Change the ('spec') to [0.13] and the ('hard') to [12]. Lastly click the ('Full Osa') button to enable it. If you render now you will notice that the plane looks like an ugly pastel color (if not, you have a different lighting setup to mine and the shader will not look the same as the images in this tutorial.) Don't worry about this ugly looking plane it will soon be a beautiful carpet.



Figure 648 Shader Screen-shot



Figure 649 Base Shader

76.3. Cloud Texture 1

Press 'F6' on the keyboard to bring up the texture panel. Click the bottommost of the long boxes to create a texture in the bottom channel. Create a new texture and rename it something like 'Red Clouds 1'. From the ('Texture Type') pull down select clouds. In the 'Clouds' settings panel change ('NoiseSize') to [0.210] and ('NoiseDepth') to [4].

Now select the Colors tab which will bring up the ColorBand for the texture. Press the orange ('Add') button to add a cursor on the colourband. Next make sure the 'Cur : 0' is showing next to the add button and change the colours as follows - (R 0.770, G 0.168, 0.168). Now click on right side of the 'Cur : 0' so it shows 'Cur : 1'. Change 'Pos' to 0.6. Set Alpha to 1 and change the colour to (R 0.732 G 0.243 B 0.243).

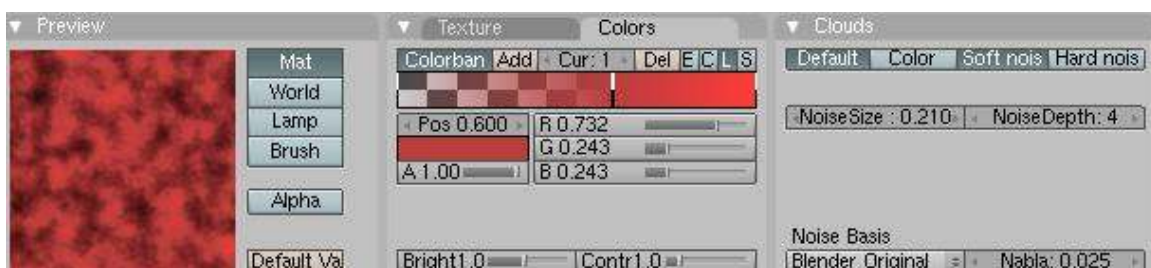


Figure 650 Texture 1 texture panel Screenshot

Now go back into the material settings and change the settings in the Map To tab as follows. Click ('Spec') twice so the text becomes yellow do the same for ('Hard'). Now select 'Subtract' for the

texture blending mode. Change ('Col') to [0.188].



Figure 651 Texture 1 materials panel Screenshot

At this stage if you render you might find it hard to notice the difference, it is apparent only where there is low light levels on the plane, don't worry about this at the moment. It means if you render with Ambient Occlusion you get a nice carpet effect of slightly varying colour.

76.4. Cloud Texture 2

Go back to the texture panel and create another texture, call it 'Clouds' and put it in the channel above 'Red Clouds'. Change ('NoiseSize') to [0.054] and ('NoiseDepth') to 4. Select 'Improved Perlin' from the Noise Basis pull down. Finally change ('Nabla') to [0.031]. Do not change any more settings here.



Figure 652 Texture 2 texture panel Screenshot

Now in the material panel, under the Map To tab change the following - Click ('Nor'). Click ('Spec') and ('Hard') twice so as they are yellow.

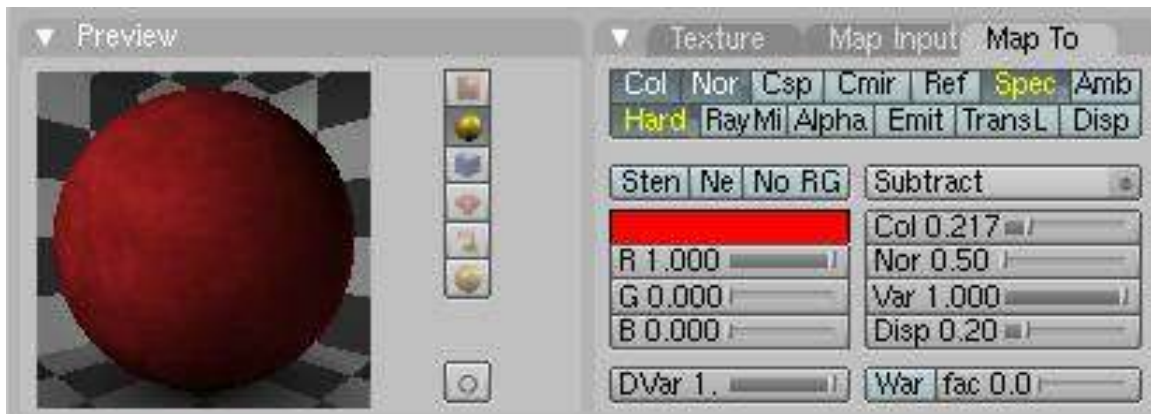


Figure 653 Texture 2 material panel Screenshot

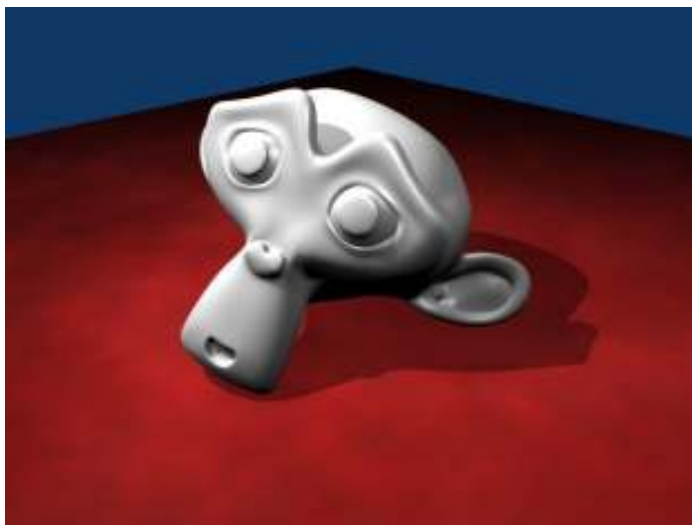


Figure 654 Render with the 2 textures so far

As you can see the material is starting to look a bit better, only 2 more textures to go.

76.5. The Final Cloud Texture

Switch to the texture panel once again and create a new Clouds texture in the next channel up. Change ('NoiseSize') to [0.010] and ('NoiseDepth') to [6]. Now click on the 'Colors' tab and change the colour of the left cursor ('Cur : 0') to (R 0.713 G 0.262 B 0.223) and Alpha to 0. Switch to Cursor 1 ('Cur : 1') and its colour settings (R 1.000 G 0.363 B 0.000) and Alpha to 1.

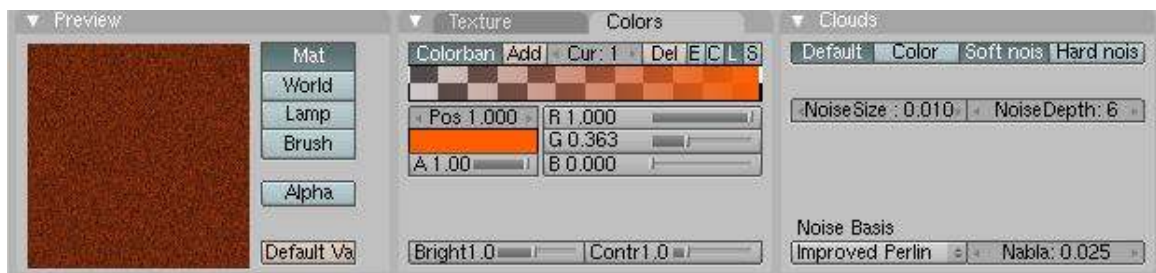


Figure 655 Texture 3 texture panel Screenshot

Now in the materials panel under the 'Map To' tab Click ('Nor'). Click ('Spec') and ('Hard') twice so as they are yellow. Leave the blending mode as 'Mix'. Change ('Col') to [0.464] and ('Nor') to [1.00].



Figure 656 Texture 3 materials panel Screenshot



Figure 657 Render of the 3 textures so far

76.6. The Last Texture

Finally, go to the textures panel one last time. Create a new Stucci texture in the next channel up. Change ('NoiseSize') to [0.006] and ('Turbulence') to [10.94]. Click on the 'Colors' tab. The first cursor 'Cur : 0' should be black with alpha 0. 'Cur : 1' should be red (R 1.000 G 0.000 B 0.000) with alpha 1.

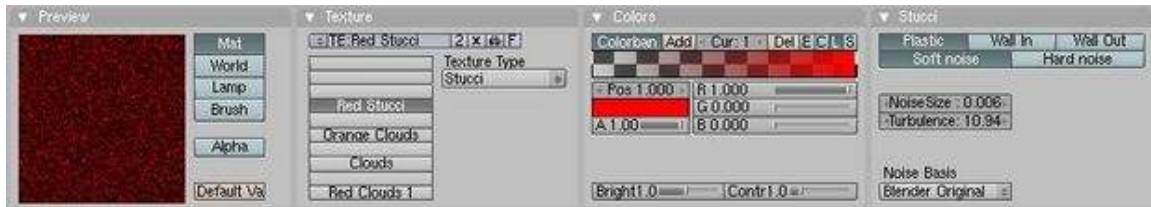


Figure 658 Texture 4 textures panel

Now go to the materials panel. Under the 'Map To' tab Click ('Nor'). Click ('Spec') and ('Hard') twice. Change the Blending type to 'Subtract'. Change ('Col') to [0.056] and ('Nor') to [0.50]. And that's it. When you render now, you should have a nice-looking carpet material. By tweaking with the colours you can create any colour of carpet.



Figure 659 Texture 4 material panel



Figure 660 Final Render of Carpet Material

77. The Rusty Ball

Making objects with image textures is not really hard for simple objects like balls, cubes, and tubes. I'll show you how to do this:



Figure 661

- Make a new Scene in Blender and delete the default cube
UNKNOWN TEMPLATE B3D:N2P/Key
Xkey
.

- Make an object you want to have the image on (I recommend a Mesh plane, sphere or tube).

(If you are making a Mesh Plane, change your view to above, by pressing

UNKNOWN TEMPLATE B3D:N2P/Key

Num7

)

To make a rusty ball I suggest an icosphere with a size of 2 which will fit nicely in the camera view.

- Sequence to add the Ball is:

UNKNOWN TEMPLATE B3D:N2P/Key

Spacebar

, Add, Mesh, Icosphere.

- Go to the materials (

UNKNOWN TEMPLATE B3D:N2P/Key

F5

) and select "O Material" from the popupmenu by the "Add new" button. (the little arrows)

- Now go to the textures (

UNKNOWN TEMPLATE B3D:N2P/Key

F6

) and choose "Texture Type" A drop down (or up) menu appears and you will get many options, the one we need is "Load image". Click it, then select "Load" and navigate to an image you want to use. (Note: JPGs, PNGs or TGAs are recommended for Blender. Bitmaps tend to get all screwy.)

(Noob Note: Do ¹ search for **Rust Textures** ²*if you do not have one handy.*)

- After this, you'll have to specify how your image should be applied to your object. To do this, go to the materials again, where you'll have to find the "Map input" tab (near the textures tab). If you have selected it, you'll see four buttons: *Flat, Cube, Tube and Sphere*. Select the option which meets your object best. You can see a simple preview of the different options in the "Preview" window and try the different modes.
- Render your object. If you can't see your picture well, you can try to rotate your object or select another option in the "Map input" tab.

You can also render videos onto objects using this method. Just select a movie in the "Load image" dialog and enable the option "Movie" at the textures buttons. *NOTE:* Blender ONLY works with Full Resolution video, not video which has been compressed using a codec. Most video software will allow you to export video as "**full frames**" or "**no compression**". Experiment a bit!

1 http://images.google.com/images?sourceid=navclient&rlz=1T4GGLR_enUS262US264&q=rust%20texture&um=1&ie=UTF-8&sa=N&hl=en&tab=wi

2 <httpa>

78. Procedural Eyeball

Level: intermediate

Version: 2.43

Building a better (procedural) eyeball!

Originally created by Jon McKay (Ammusionist) and posted on blenderartists.org/forum <http://blenderartists.org/forum/showthread.php?t=85722&highlight=eyeball>¹ The end result of this tutorial is an eyeball that fulfills the following requirements:

1. Single mesh
 - I wanted to be able to append a single object into any project.
2. Procedural Textures
 - I didn't want to have to rely on image maps that could get lost.
3. Versatile
 - One single model to be used for any type of character, be it human, alien or whatever.
4. Easy
 - Any effects needed to be quickly accessible. I don't want to be wasting time faffing around with colour-band settings at the texture level
5. Impressive
 - This sucker needs to look good any way it goes.
6. One other thing I really wanted was the iris musculature to follow the pupil dilation!

[Total Noob] I found this tutorial both informative and useful. It's best to download the PDF version with the screen shots if you want to get as much info out of it as I did, and in fact, understand all that's going on in the Tute

New Link (08th Nov '11): Here² is the full tutorial as a pdf file. 9 pages of screenshots and extremely detailed instructions. (1.2mb)

(Noob note: In my blender 2.49, I had to unclick the 'Traceable' setting on the cornea of my eye when following the PDF tutorial, else I ended up with a giant black 'olive', this does not appear to affect ray mirroring, and corrects my problem.)

1 <http://blenderartists.org/forum/showthread.php?t=85722&highlight=eyeball>

2 http://www.aurellem.com/aurellem/Eyeball_Tutorial.pdf

78.1. Creating the Mesh

[noob note1] Maybe worth to look at this link instead... <http://web.pdx.edu/~wlf/tut.html>³

[noob note2] Broken link.

Note: You might want to check out this video first: <http://www.youtube.com/watch?v=KKmHUZroaBA>⁴

The objective here is to make a great looking eyeball. That means it may not be anatomically perfect, but there are some things we need to take into account: The eye should consist of a white (white bit with veins); an iris (coloured bit); a pupil (black but in the centre of the iris) and a cornea (clear bit surrounding the whole eye). These may not be absolutely correct, but these are how I'm referring to the pieces of the eye.

- Start with a ball: Go to a front view (Keypad-1) and create a UV sphere. 24 Rings and segments should be sufficient. It'll give us plenty of smoothness and enough verts to work with when we create shapes.

NOOB NOTE: If you want the eye to point toward you, you may have to rotate it 90 degrees. Go to side view [num3], click Rkey and rotate while pressing the control key. Return to front view.
(Noob Note #2: If you, like most people, want the eyeball's iris and pupil to point towards the front, but don't want to change views, press the RKEY, then XKEY to restrict rotation to the X-AXIS, then, while holding CTRLKEY, rotate the sphere by 90°.)

- This sphere will become the coloured parts of the eye, but we also need a second sphere for the cornea. Use the Z key to switch to a wireframe view (if you're not already there). Hit the A key a couple of times to ensure all points are selected then Shift-D to duplicate the sphere.

(Mast3rlinkx's addition: I started the other tutorial, Realistic Eyes in Blender, but it's incomplete. I thought I could use the base mesh of that tutorial here for a more realistic look, and a less confusing mesh. I used this shape for the cornea.)

- Press S-Key to scale and scale the second sphere to slightly larger than the first. (Hint: Hold shift while scaling for fine adjustment) or you could also enter a value, 1.01.
- Now that we have a nice confusing mesh, we're going to make our life easier. We're going to hide the outer bit for the moment. Hit the "H" key.
- Don't panic like I did the first time I accidentally hit H, you can bring it all back with Alt-H. (Note: In blender Alt + some key often reverses the effect of pressing the key in the first place!)
- OK Now to create the basic mesh for the inner part. Basically, we need to make a concave iris/pupil section.
- Switch to a side view and use the Border Select (B-Key) to select the first four rings and the tip. Now we're going to flatten these. Press "S" for scale "Y" for Y axis only and "number 0" for scale-to zero. (Mast3rlinkx says: If you do this, your poor eyeball will have a pinched appearance where the pupil and iris should be. What you *should* do is hit S, then Z, then 0. Then drag along the Z-Axis so the flattened area is flush with the curve of the sphere.) (Noob Note: The axis you

³ <http://web.pdx.edu/~wlf/tut.html>

⁴ <http://www.youtube.com/watch?v=KKmHUZroaBA>

need to use will change depending on the way you aligned your eye. If you want the front to point along the Y-axis, scale Y to 0; otherwise, scale Z to 0.)

- Now grab the flattened verts with the "G" key and press Y so that it goes the right way, and drag it a little into the eyeball. [noob note: you can also select two opposite vertices in the fourth ring before scaling along the Z axis, and press CTRL + S to snap the 3D cursor to the center of the fourth ring. Then set the pivot point to 3D cursor.]
- Now for the cornea. It's not perfectly spherical; it bulges slightly at the front. We're going to use the Proportional Editing tool.
- First, we need to un-hide the cornea. Hit Alt-H
- Usually you would use the O-Key, but this little menu item allows you to activate proportional editing for Connected vertices.
- Once activated, you can select the type of falloff from the 3D header bar. Select "Root Falloff" (Mast3rlinkx's Second Addition: Instead of root falloff, which looked less realistic than it could have, I used Sphere Falloff and got a more realistic look. That's just my opinion, being someone who points out the anatomically incorrect portions of a cartoon character's appearance.)
- Press A-Key a couple of times again to make sure no points are selected, switch to a side view (Keypad-3) and right click just to the left of the front-most point. We just want to select that one vertex.
- Now comes some fun. We want to drag that point away from the eye in the y axis, so press "G" for grab and "Y" for Y axis. As you move the vertex, you'll see a circle that defines the influence of the proportional editing. Adjust it so it is about $\frac{1}{4}$ the height of the whole eye and drag it out until it looks right.
- To clean the model up, we're going to apply a subsurface and apply some smoothing. Go to the editing buttons and find the modifier panel.
- Add a subsurface modifier and set both level and render level to 2.
- Now hit Tab to switch to Object mode and hit the "Set Smooth" button in the Link and Materials panel.
- Now would be a good time to save your work before we go on to textures.

78.2. Adding the Textures

You're going to need multiple materials on this mesh. It's possible to apply materials to selected faces.

- First, switch to editing mode and change the select mode to Face select. (Oh, and switch off proportional editing if it's still on from the last section)
- Press A a couple of times to ensure nothing is selected.
- Press Z to switch off wireframe; this will be easier without it.
- Place the mouse pointer in the centre of one of the faces on the outside layer of the eye and press "L" to select linked faces. If you get a message saying "Nothing Indicated" try positioning the pointer on a different face and try again.
- Once the cornea section is selected, in the "Links and Materials" panel of the editing buttons, click "New". The counter above it will change to "1 Mat 1" best read as material 1 of 1. Click "Assign" to assign the cornea faces to material 1.

- Hide the cornea faces (H-Key). We'll assign the rest from the outside in. If a face is already assigned to a material and it is assigned to another, it just switches place. Therefore, it's easiest next to assign all the visible faces.
- Press A to select all and, as above, click “New” then click “Assign”.
- Now for the iris and pupil. Hit A to deselect all.
- Switch to a front view and zoom in. Hit “B” twice to enable paint select and select the central faces plus one ring out. You can control the size of the selector by rotating your mouse wheel.
- Again, create a “New” material index and “Assign” the pupil faces to it.
- Finally, click A to deselect all and use paint select again to select the remaining faces on the iris. Assign them to a new material.
- Now we're ready to make the materials. Switch to object mode (Tab). Change to the object editing buttons and in the Draw panel, locate the “Transp” control. Set this on. This means as we go we'll be able to see through the cornea.

78.2.1. Cornea Material

We'll start with the simplest texture first. The cornea is basically just colourless and transparent. Note that we're using Z-Transparency instead of Ray Trans. This is because some lighting conditions make the eye under the cornea difficult to see. (NOTE: Until I get the images uploaded, you should refer to the pdf version for the settings. Meanwhile, here is how I see them:

Go to the shading buttons (F5)

Under the Links and Pipeline tab, ZTransp is clicked along with Radio and Tracable.

Under the Material tab, col is white and alpha is 0.1 .

Under the Shaders tab, CookTor Spec is 1.674 and Hardness is 333.

Under the Mirror Transp tab, Ray Mirror is clicked, RayMir is 0.4, Depth is 3 --Bonapon⁵ (talk⁶) 14:31, 6 October 2009 (UTC))

(Noob note: In my blender 2.49, I had to unclick the 'Traceable' setting on the cornea of my eye when following the PDF tutorial, else I ended up with a giant black 'olive', this does not appear to affect ray mirroring, and corrects my problem.)

78.2.2. White Material

The white of the eye's a little trickier. The material here includes some red veins that can be seen at the side of the eye, but not from the front. This can be done with two textures, but there's a cheat we can use.

First, the veins:

⁵ <http://en.wikibooks.org/wiki/User%3ABonapon>

⁶ <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

Switch to Material Index 2 (In the Links and Materials panel, click the right arrow so it reads “4 Mat 2” or Material 2 of 4).

Click the cross or X next to the material name and add a new material.

Go to the texture buttons:

Using a marble texture, copy these settings.

(Here are the settings that I see

Under the Colors tab:

Cur: 0: Pos=0; R=1, G=0.617, B=0.629; A=1;

Cur: 1: Pos=0.340; R=1, G=1, B=1; A=0;

Under the Marble tab:

Sharper, Soft noise and Sin are clicked;

NoiseSize=0.634; NoiseDepth=5; Turbulence=26.04;

--Bonapon⁷ (talk⁸) 15:25, 6 October 2009 (UTC))

The two colour band items are shown here separately. This makes a nice marble texture that will evenly cover the eyeball. So how to hide it from the front of the ball? Another colour band trick, but this time in the material itself.

Here they are in all their glory.

(As I see them:

Under the material tab, Col is white and alpha is 1;

Under the Ramps tab:

Cur: 0: Pos=0.690; R=1, G=1, B=1; A=0; Input is Shader; (These settings are not shown in the pdf so I have extrapolated. Please correct if wrong.)

Cur: 1: Pos=0.690; R=1, G=1, B=1; A=1; Input is Normal;

--Bonapon⁹ (talk¹⁰) 15:43, 6 October 2009 (UTC))

The colour band items are both white with one alpha 0.0 and one alpha 1.0. As you can see they are very close together to give a fairly clear line that the veins will stop at. The input setting for the band is set to “Normal” this means the left hand side of the band refers to faces that are parallel to the camera view and the right is faces that are facing directly towards the camera.

I've also added a bit of mirroring so that we get a little scenery reflected!

There's also some colour band fun to be had with the pupil material.

⁷ <http://en.wikibooks.org/wiki/User%3ABonapon>

⁸ <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

⁹ <http://en.wikibooks.org/wiki/User%3ABonapon>

¹⁰ <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

78.2.3. Pupil Material

I wanted a nice black texture, but I also wanted to get a red-eye effect if a light is shone directly at it from behind the camera. Why? I guess just because I can, and now so can you!

As before move to the next material index and create a new material.

There's no textures in this one. It's basically black and a little reflective with a nice hard specular.

(In the materials button (F5):

Under the Links and Pipeline tab, Ztransp, Radio, Tracable and Shadbuff are clicked;

Under the Material tab, Col is Black, presumably with an alpha of 1, while Spe is white with an alpha of 0.6;

Under the Shaders tab, CookTor spec is 0.594, Hard is 294;

Under the Mirror Transp tab, Ray Mirror is clicked, RayMir is 0.3;

--Bonapon¹¹ (talk¹²) 18:14, 6 October 2009 (UTC))

Have a good look at the colour band settings though!

Position 1 is black with an alpha of 0 and positioned at 0.97. Trust me on this setting, it took a while to get it right!

(Under the Ramps tab:

Cur=0; Pos=0.970; R=0, G=0, B=0, A=0; --Bonapon¹³ (talk¹⁴) 18:22, 6 October 2009 (UTC))

The second is all the way across at position 1 and is fully red (with no blue or green) and alpha 1. We want the red-eye to respond to light not angle like the white of the eye, so set the colour band input to "Shader".

(Cur=1; Pos=1; R=1, G=0, B=0, A=1 (The alpha of the second position is not shown in the pdf) ; --Bonapon¹⁵ (talk¹⁶) 18:22, 6 October 2009 (UTC))

Now the bit you've all been waiting for – the iris!

78.2.4. Iris UV Map

Save your work and grab a cup of coffee for this part. It's a little involved.

We need the texture to stay glued to the mesh irrespective of the shape of the mesh. This gives the effect of the muscles in the iris expanding and contracting. In a nutshell, we're going to UV map a procedural texture to a single part of the mesh.

¹¹ <http://en.wikibooks.org/wiki/User%3ABonapon>

¹² <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

¹³ <http://en.wikibooks.org/wiki/User%3ABonapon>

¹⁴ <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

¹⁵ <http://en.wikibooks.org/wiki/User%3ABonapon>

¹⁶ <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

OK, here we go. Switch to a material editing window. You'll need to be able to see both a 3D window and an image window. It wouldn't hurt to see a buttons panel as well. My material screen kind of looks like this.

(Noob Note: *To clarify: an "Image Editor" window option is in the **Select Window Type** dropdown button situated at the very left of the header next to the **View** dropdown. Select '**UV/Image Editor**'. To achieve the screen the author describes, first split your view. Then select a front ortho view with **NUM1** in the right window. In the left select **UV/Image Editor** from the **Select Window Type** button as described.)*--*WithAlligators*¹⁷ (*talk*¹⁸) 05:59, 24 February 2010 (UTC) (Blender 2.49b)

Make sure the window is displaying a front view (Keypad-1) this will be important when we unwrap the bit we're going to use.

Change the 3D window to mesh editing / face select and use the brush select (B-Key twice) to select the iris faces.

Now change to UV Face Select mode. The faces should remain selected. (As of Blender 2.46 the UV Face Select mode no longer exists. It's been merged with Edit mode. All you have to do is select all the vertices/faces/edges in Edit mode and press the U-key.)

With the cursor in the 3D window, press "U-Key" for unwrap and select "Project from View". The image window should now contain a nice even circle of dots. Put the cursor in the image window and press "A" to select all the points. Scale and move them until they fit nicely in the outer square of the image grid. Now select each ring in turn using brush select, starting from the innermost, and scale them until they cover from the centre to the outside of the image grid as shown.

Now we've got a map, lets get a texture. Switch back to a modelling screen and change to object mode for the next part.

78.2.5. Iris Material

Firstly, a texture to control the colour blend across the iris. Generally it starts on the outer edge dark and gets lighter as it moves to the centre. Sounds like a job for the blend texture.

The texture itself is fairly straight forward, but again, the colour band is where the magic happens.

I've pictured all three stops in the colour band. Note the alpha values. This blend texture is meant to tint the iris texture so it's lighter in the centre and darker in the outside. **(Note:** I will have to create my own screen prints as I don't know if the original author has uploaded images or if they are copyrighted. For this to happen, I will first have to complete the tutorial myself.)--*Bonapon*¹⁹ (*talk*²⁰) 20:34, 8 October 2009 (UTC))

Now for that iris texture:

I stumbled over this texture almost by accident when I was working on a different project. Copy the settings as they appear here.

¹⁷ <http://en.wikibooks.org/wiki/User%3AWithAlligators>

¹⁸ <http://en.wikibooks.org/wiki/User%20talk%3AWithAlligators>

¹⁹ <http://en.wikibooks.org/wiki/User%3ABonapon>

²⁰ <http://en.wikibooks.org/wiki/User%20talk%3ABonapon>

Note: the colour band points are black with alpha 0 and white with alpha 1. This allows the material colour to come through the texture, so we can set the main eye colour in the material buttons.

The main colour is blue which sets the colour of the iris. The centre colour comes from the texture settings:

The blend texture is applied to the UV map input. The green colour is used as a lighten mix to tint the iris near the centre. Using the Stencil option allows the other texture to show through.

The iris texture is applied to UV again and some normal as well as colour is applied. This makes some small shadows in the iris when the light is from the side. Add some lights and have a look at your eyeball so far.

78.3. Pupil Dialation

We're going to use shape keys to create the pupil effects.

We're going to create a normal eye and a cat's eye. Fortunately, the fully dilated eye is the same in both cases so that's going to be our shape key basis.

Go to the edit buttons and select the Shape Key panel.

Make sure "Relative" is selected and click "Add Shape Key" to create the base shape.

Switch to edit mode, point selection. Select just the first loop in of the iris. To select an edge loop in point mode, place the cursor on a line between two points that's on the loop and Alt-Right Click. Shift + Alt-RClick will add the selected loop to what's already selected.

Now scale this to just inside the width of the loop that the iris shares with the white..

Now do the same with each of the other two loops. This should produce a nice dilated pupil.

Hint: If you scale the inner loop out, but the edge of the colour doesn't follow it exactly, drop it and hit Shift-E, then move the pointer to the right. This causes the subsurfaced mesh to closer follow the selected edge.

Now to create the normal (human) pupil shape. Switch to object mode (Tab) and add a new shape key. Name it "Normal Pupil Contract"

Now grab the edge loop between the pupil and the iris. Scale is down so it's just a little bigger than the loop in the pupil. Select all the points in the pupil as well now and scale them all very small.

Now select the next loop out and scale it down, followed by the final edge loop. It should look like this.

Now switch back to object mode and try the shape key slider out. Pretty neat huh?

Let's make a cat's eye now. Switch shape keys back to "Basis" and add a new shape key called "Cat Pupil".

Use the same method as before to scale the edge loops. As you scale, restrict your scaling to the X axis (That is "S" then "X"). You should end up with a shape key like this:

That's pretty much it. Try some other things like rotating iris edge loops in the shape keys for spiral eyes, or weird shapes for the contracted pupil.

Don't forget to save – and enjoy!

(noob note: when switching between sliders, the previous one must be reset, or their effects will combine in weird and unfathomable ways)

79. Creating Pixar-looking eyes in Blender

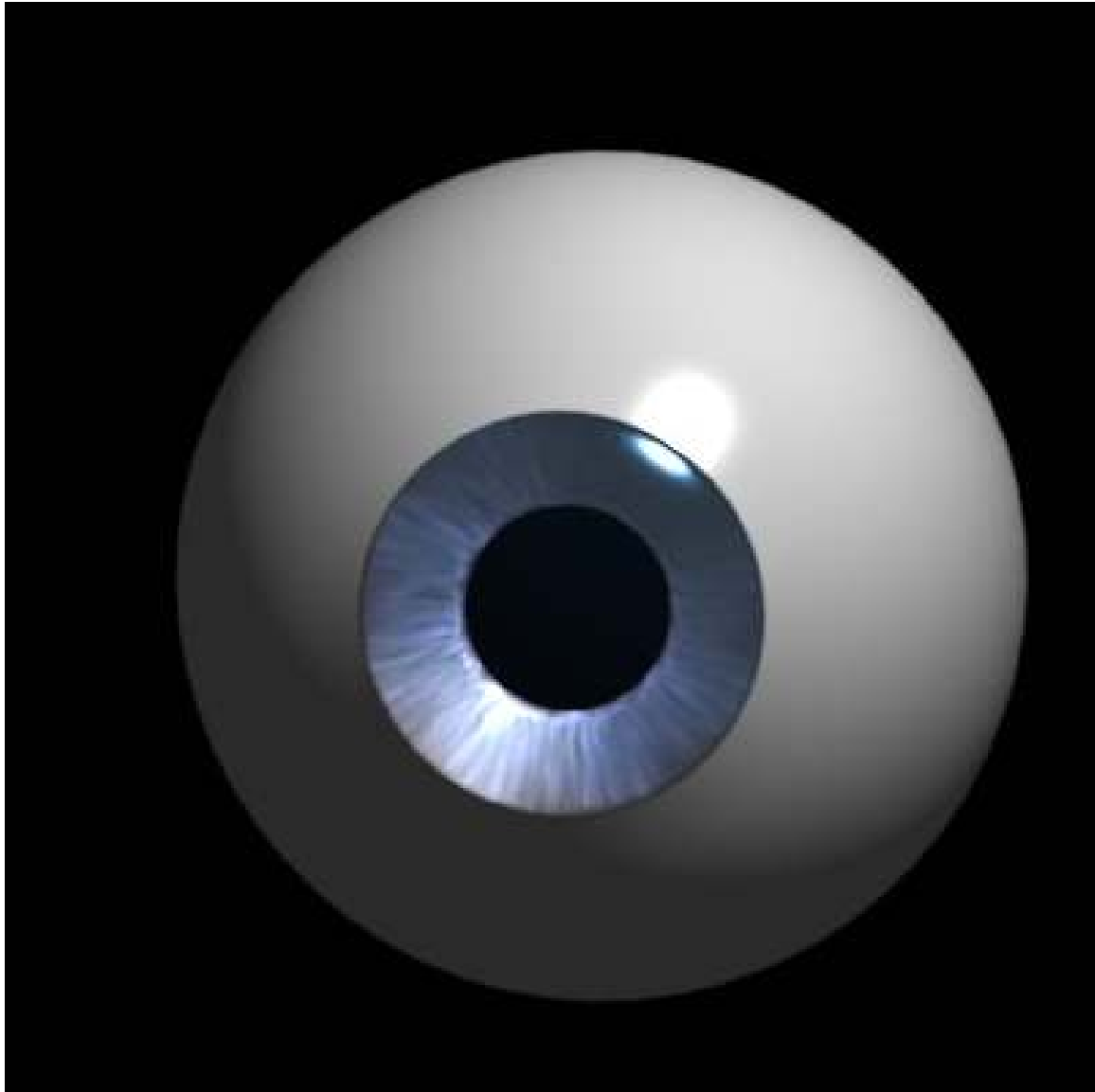


Figure 662 The final result, with a blue iris.

Note: This tutorial uses the same modelling and texturing technique described in the well-known MAX tutorial by Adam Baroody (<http://www.3dluvr.com/rogueldr/tutorials/eye/eyes.html>).¹ The

¹ <http://www.3dluvr.com/rogueldr/tutorials/eye/eyes.html>).

sole purpose of my tutorial is to make this technique more popular among the Blender users by explaining how to achieve the same result with Blender.

The goal of this tutorial is to make a Pixar-looking eye. One of the main reasons that Pixar's characters really convey life is in their eyes. They have depth, you can see how the eye not only shines but it "collects" light. You may think that you can't achieve this effect without raytracing but you're wrong. The secret of this depth is in the modelling of the eye. Let's see how it works!

79.1. Parts of the Eye

In this picture you can see the "ingredients" of the eye model. The blue mesh at the left is the cornea. Its shape allows for a small spot of specular light to appear on it even if the light is in a far side position. The mesh next to it is the iris. Now notice how it's a bit concave. That's the tricky part - the shape of the iris allows for a wide soft specular light to appear at the opposite side of the lamp direction. This fakes refracted light from the cornea and makes the illusion of "collecting" light and creates depth. The next mesh is the eye pupil - a simple circle. The pupil size is the same size as the iris hole. You can position it close to the inner side of the iris. And finally - the eyeball. It's a simple sphere with a hole in it.

I won't go deep into modelling of each element - it uses Blender's subdivision surfaces and it's quite simple as you can see.

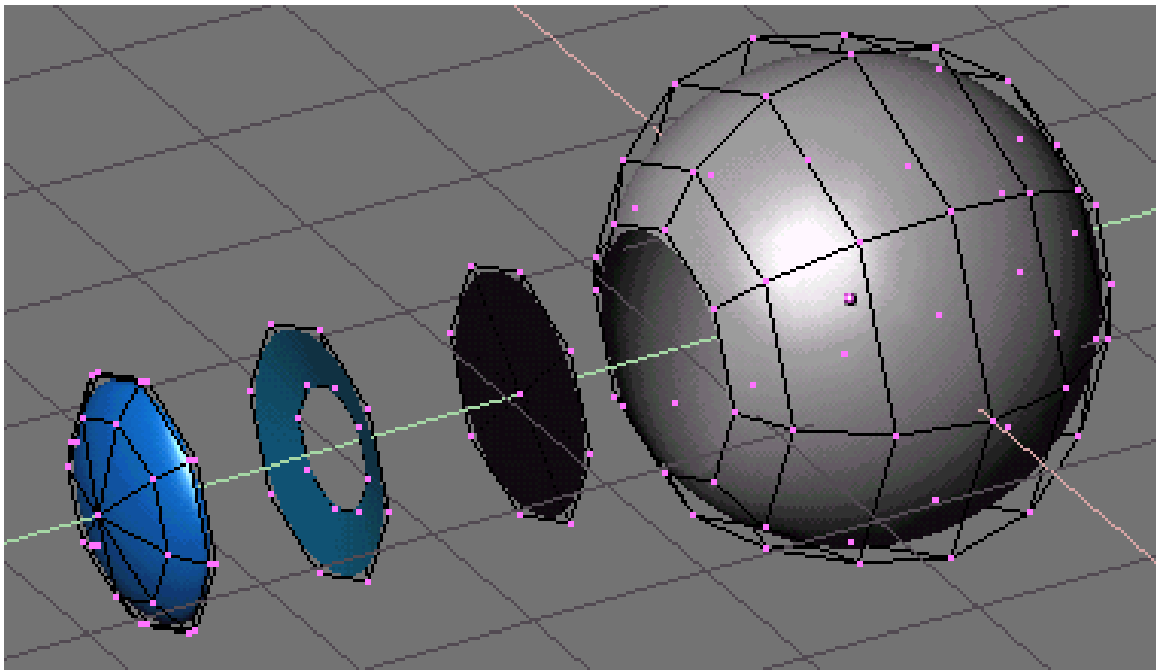


Figure 663

79.2. Materials

Now let's look at the materials.

79.2.1. Sclera

To make the eye white, press NUM3 to go into side view and add a UVsphere with 8 segments and 8 rings. To create the hole at the front of the eyeball, delete the 8 triangular faces that make up one end of the eyeball. Ctrl+Tab+3 to go into face select mode, B for border mode and select the middle 8 triangles as shown below, press Delete or the X button and LMB on Faces.

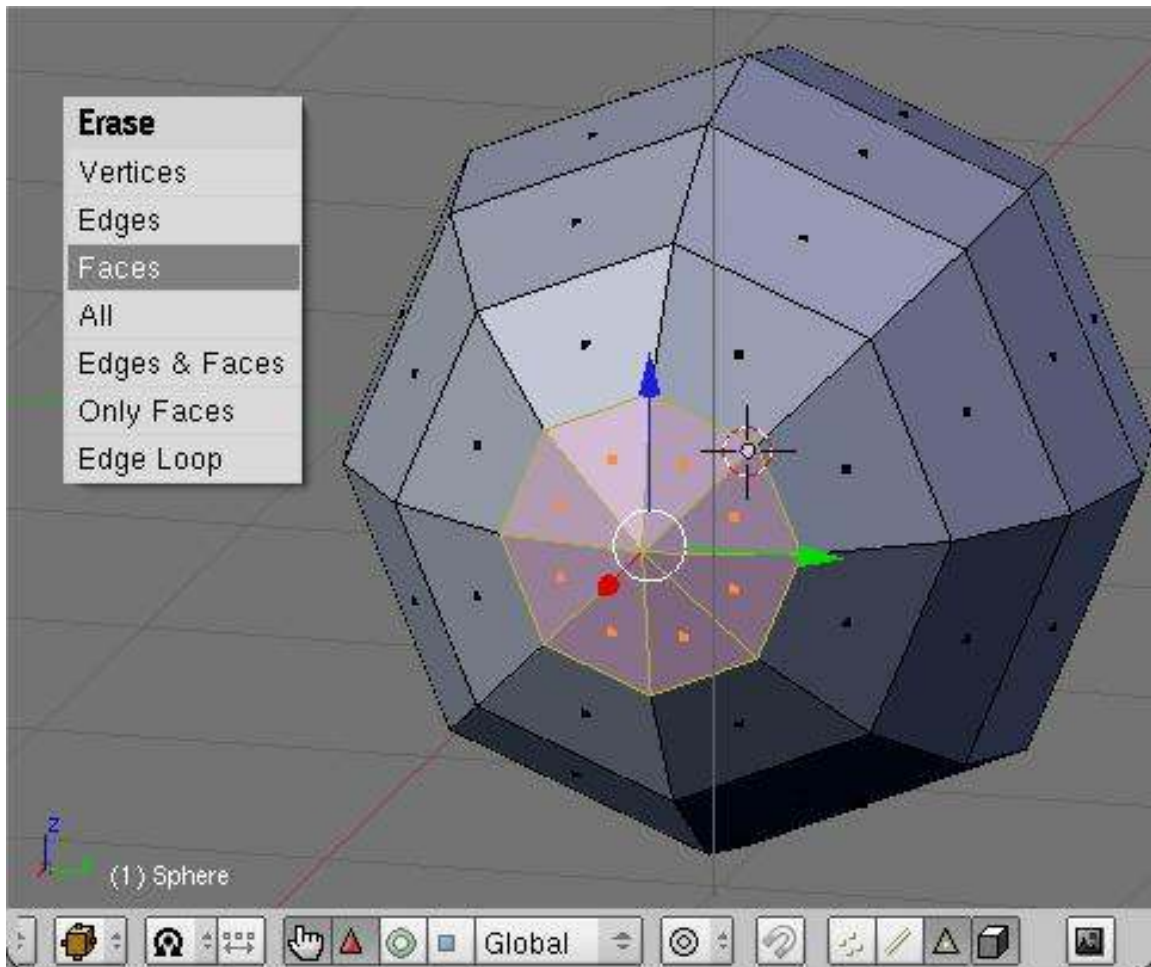


Figure 664

The resulting hole will be ringed by 8 vertices (which you could subsequently extrude and use to help you model the sections below - just a suggestion for those who feel capable)

The eye white² has white color and high values for Spec and Hard (depends on the lighting), as well as a moderately high Ref value (0.700 - 0.800). Optionally you can use a reflection map to make it look more wet but I usually don't do this.

(Noob note: Using a Phong specular shader turned to work quite well.)

(User: Make sure shadow buttons are off or it will black out the iris. Spec and Hard are located in the Shaders tab.)

79.2.2. Iris

- Loop select the opening of the eye (ALT+RMB click edge of opening).
- Duplicate the loop (SHIFT+DKEY). Move the new loop away from the eye a bit perpendicular to the opening.
- With the loop still selected, extrude (EKEY). Without moving the extrusion, press ENTER.
- Resize by half (SKEY then .KEY then NUM5 then ENTER).
- Move the inner loop back toward the eye a bit to form the concave shape.
- Create a new material³ and create a new texture⁴ called iris. Then hit F6 or click the spotty square icon

2 <http://en.wikipedia.org/wiki/Sclera>

3 Chapter 36 on page 213

4 Chapter 40 on page 235

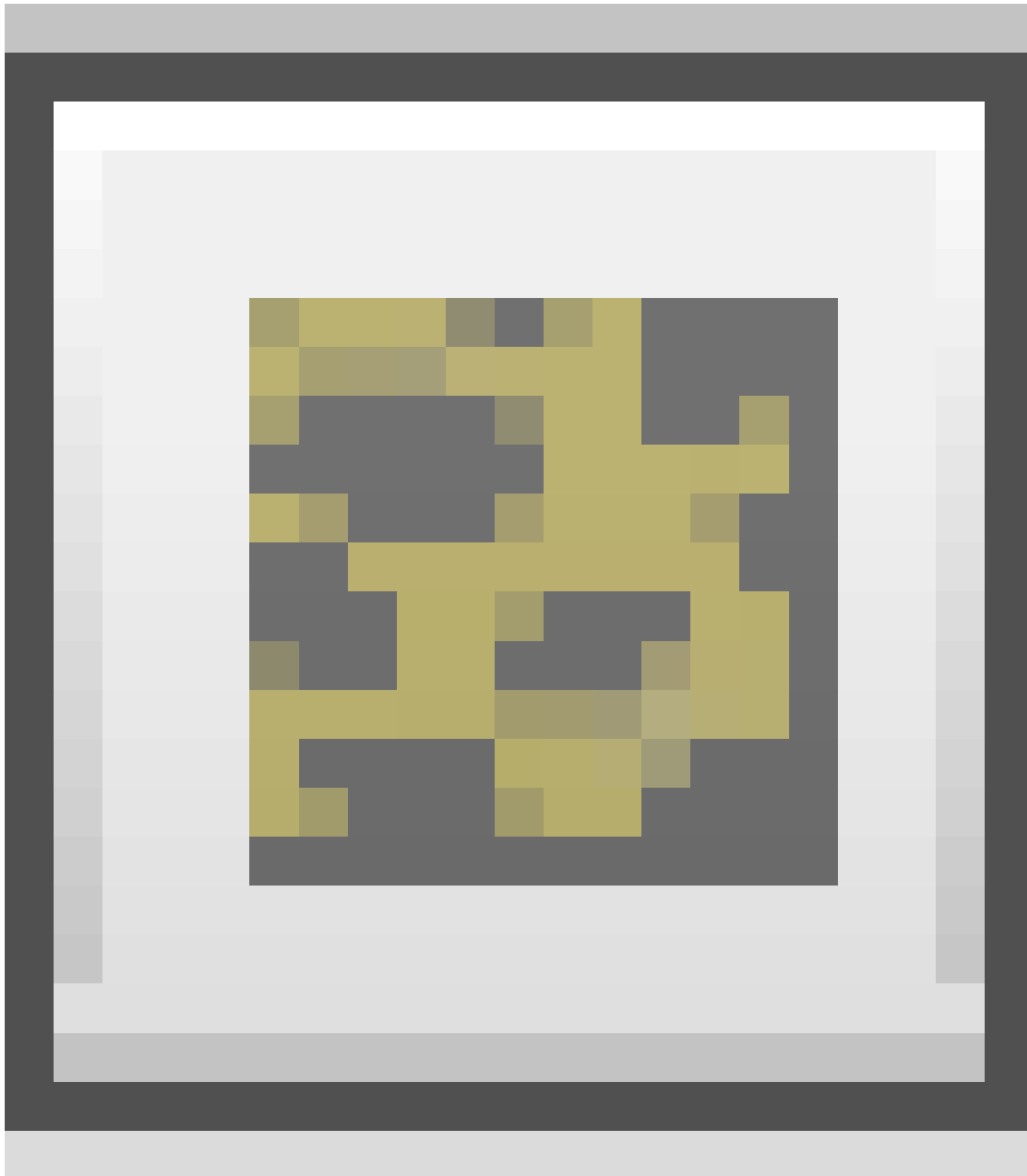


Figure 665

to open the texture panel, choose Image as the texture type and then press Load Image button and use the image below.

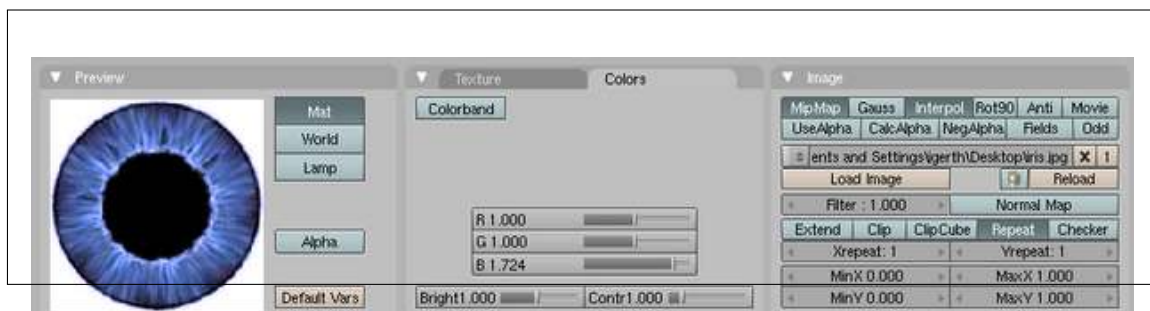
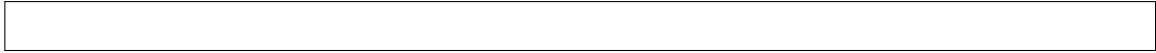


Figure 666 Setting the iris color.



Here is the eye texture taken from the picture above



Figure 667 The iris

All that is required now is a bit of tweeking of the texture size using the Xsize and Ysize values in the Map Input tab, and scaling the pupil hole size in Edit mode. Also please note to change in the map input settings from 'flat' to 'cube'. You can tweak the RGB values and brightness/contrast of the image to achieve the appearance you want. Use a smaller value for Hard (about 50) otherwise you'll have a too shiny look instead of soft specular that fakes refracted light. The Spec value depends on the energy and distance of the light that illuminates it. Generally you'll need to take care that the refracted lighth on the iris should be no more than half as bright as the small specular spot on the cornea - otherwise you'll achieve the bad effect of two specular spots. Oh, another important thing -

join the four meshes before tweaking the texture coordinates. Otherwise you'll have to do the job twice after you join them, because the texture space is changed. And activate shadeless button.

79.2.3. Pupil

The material for the pupil is a simple black color with the "shadeless" button on.

79.2.4. Cornea

The cornea uses a transparent material (**alpha = 0.1**) with **Spec = .6**, **Hard = 255** and **SpecTra = 1**. "**Ztransp**" should also be turned on (found under Links and Pipelines tab). The cornea is simply a piece that fits exactly in the middle of the hole in the eye white. Make sure that the 'Traceable' button, under Render Pipeline, in the Links and Pipelines tab, is switched off. Noob note: I found it helps to turn off Shadebuff since, as far as I know, you don't generally want the cornea to cast a shadow.

Noob question: I don't know where is the "SpecTra" Value. I'm working on 2.46. Thanks a lot. You can find "SpecTra" under "Mirror Transp"

79.3. Lighting

The lighting is simple - move the eye to a new layer, create a new lamp and make the lamp affect only this layer. Position the lamp at a good angle so you have a small shiny spot of specular light on one side of the iris and a soft spot of "refracted" light on the other side. You can use a backlight to prevent the eyeball from being too dark at the non-illuminated part.

To move the newly created lamp to a new layer, press SHIFT+M and select the second blue button and press OK. P

Noob Question: Why do you need to create a new layer?

That's it! Now you're (almost) ready to start with character animation. You have a nice eye, now you only need a character for it!

79.4. Changing the Eye Color

You can change the color of the eye by either changing the cornea color, or by changing the iris color itself. Changing the cornea color might be like putting on colored contacts.

79.4.1. Changing the Cornea Color

To change the cornea color, do the following:

1. In the mesh which is used to create Cornea set the value of the alpha slider to 0.2 (or more if you desire)
2. Change the color of the mesh to whatever color you like

3. Turn off traceable

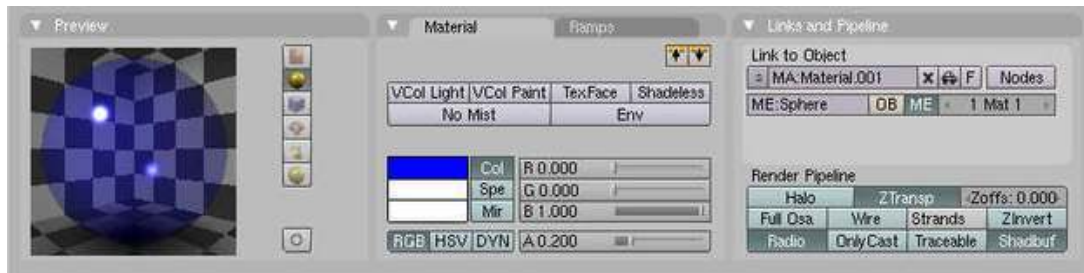


Figure 668

4. Render!

(and remember to smooth out everything by selecting the particular mesh going to edit buttons and clicking on set smooth)

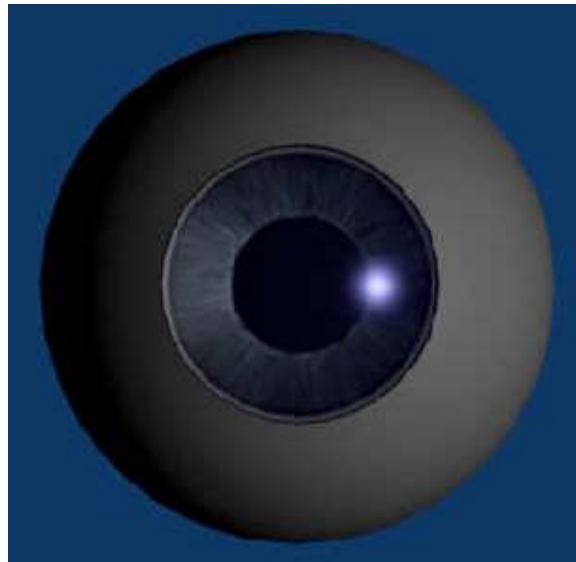


Figure 669

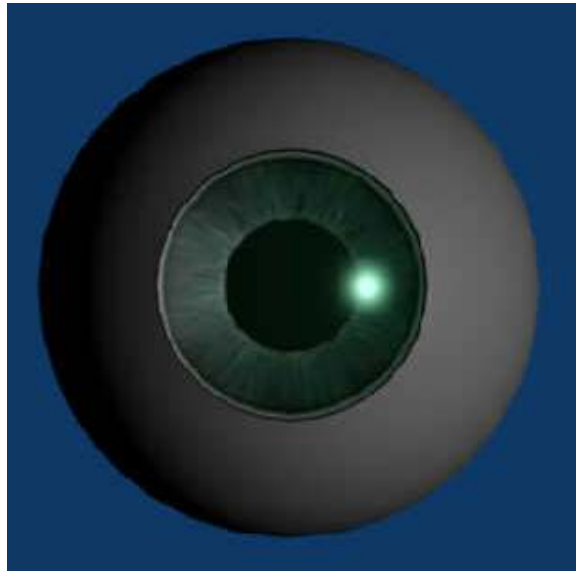


Figure 670

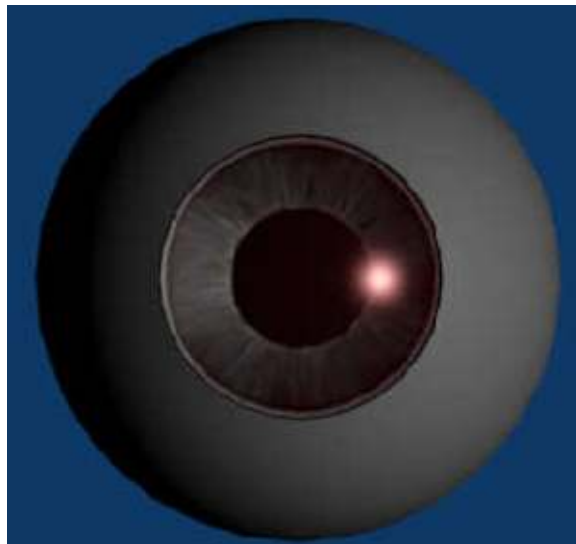


Figure 671

79.4.2. Changing the Iris Color

To change the iris color you can either edit the iris image external from blender or you can modify the RGB values of the image when imported into Blender.



Figure 672 Setting the iris color.

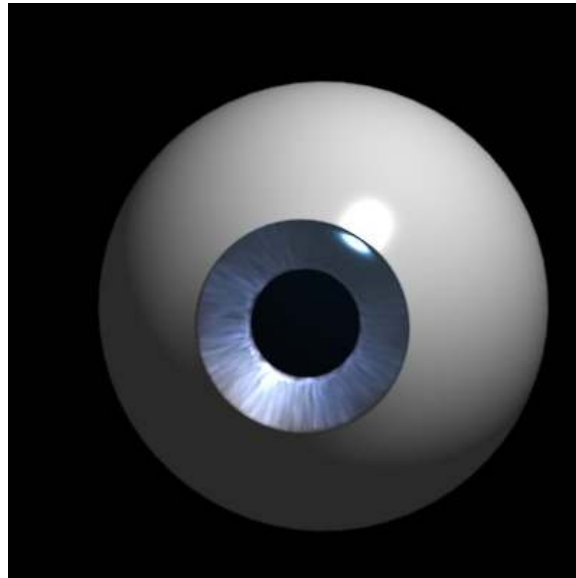


Figure 673

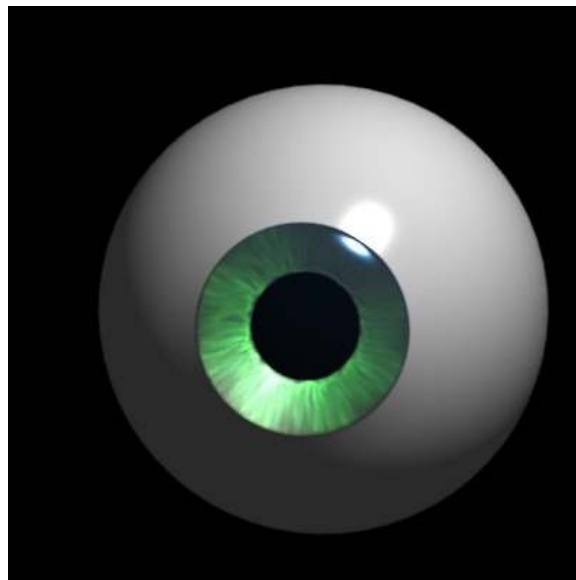


Figure 674

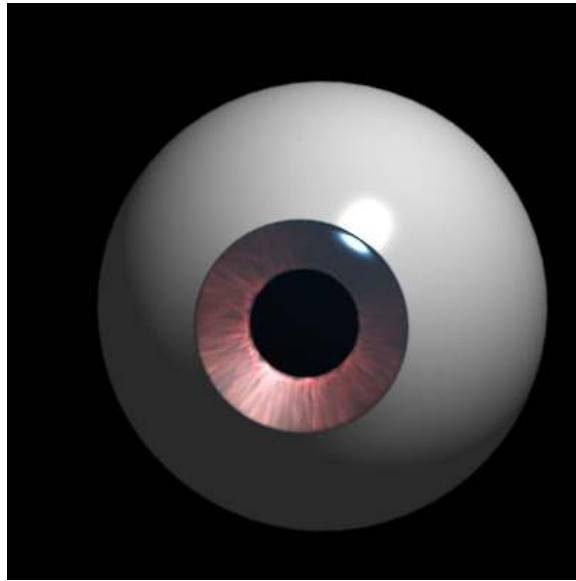


Figure 675

79.5. Pixar Eye Video Tutorial

- Ira Krakow's Blender 2.49 Pixar Eye Tutorial⁵

79.6. Useful Links

For rigging an eyeball, (making it stay in one place while turning to look at something) and other eyeball stuff, you can visit this site. This tutorial assumes you already have the eyeball made above or one of your own. Click.⁶ 5jun2009, the PDF shows no rigging, only creating an eyeball. (May 10th 2011, the site no longer exists.)

⁵ <http://forum.irakrakow.com/viewtopic.php?f=19&t=250>

⁶ <http://rjnlk94.googlepages.com/eyeballtutorial>

80. UV Map Basics

In case you're wondering, UV mapping stands for the technique used to "wrap" a 2D image texture onto a 3D mesh. "U" and "V" are the name of the axes of a plane, since "X", "Y" and "Z" are used for the coordinates in the 3D space. For example: increasing your "V" on a sphere might move you along a longitude line (north or south), while increasing your "U" might move you along a line of latitude (east or west).

Another explanation can be gleaned from the Blender manual. Imagine a paper 3D model of an object, e.g. a sphere, that is to be laid flat on a table. Each of the 3D coordinate of the sphere can be mapped to the 2D coordinate on the flat piece of paper. Blender provides another view of the vertexes(coordinates) in the UV/Image Editor. You can select and edit these 2D vertexes just like in the 3D Editor window. The purpose of this unwrapping of the coordinates is just to map these coordinates to images/pictures so that the 3D image can have a realistic looking surface with textures derived from these images.

You can watch a good video tutorial from the main Blender¹ site. It is called **LSCM UV Mapping** or (Least Squares Conformal Map UV Mapping) and it is located on this page: <http://www.blender.org/education-help/video-tutorials/modelmateriallight/>

80.1. The Basics of UV Mapping

Add an icosphere

We'll use a sphere for this demonstration. Make sure you are in top view so the equator of the icosphere we are creating will be parallel to the x/y-plane and unwrapping will give good results (NumPad7 or View → Top). Delete the initial cube (x). Then, create a new model, an icosphere (SPACE → Add → Mesh → Icosphere). Leave the settings at default: subdivisions 2, radius 1.0

Noob note:

1. *This tutorial is actually an advanced method of unwrapping and is almost complete. It is also straight forward in that, you use actual pictures of 3D objects without any distortions. Unfortunately, NASA and panoramic pictures tend to come in the form of the 360x180 degree format. Blender provides automatic transformation of the 3D coordinates of a sphere into its 2D versions, i.e. these 360x180 degree formats. At the end of this tutorial, there are steps for an alternative method of mapping earth pictures to a 3D sphere.*

¹ <http://www.blender3d.org>

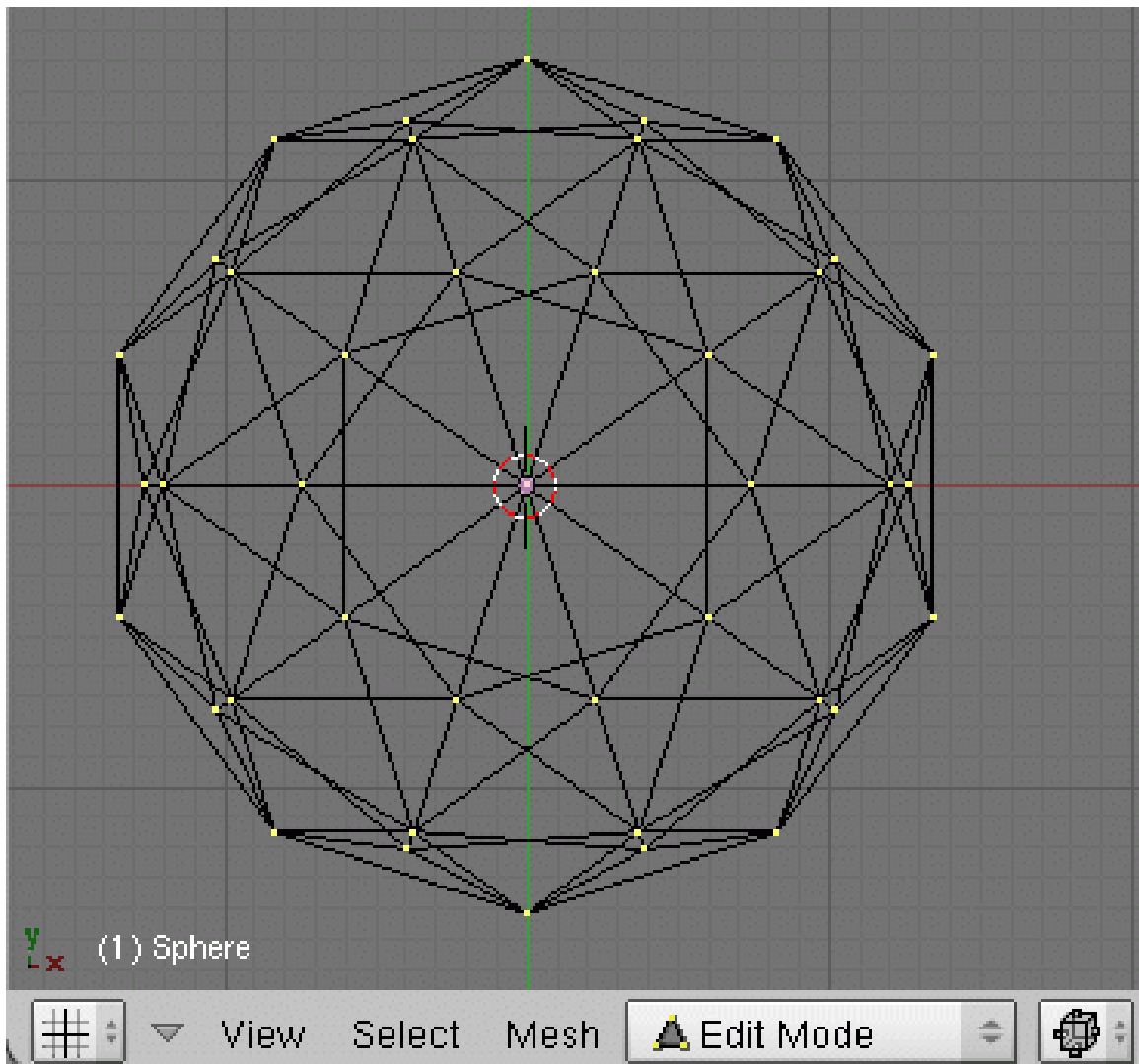


Figure 676

Mark a seam

In side view, select a ring of vertices on the icosphere. (Like an equator). This can be done easily by going to a side on view (NUM1 or NUM3) and drawing a selection box around the middle row of vertices (BKEY + LeftCLICK+drag). Make sure Limit Selection to Visible is NOT enabled first.

(Limit Selection to Visible appears to be labelled "Occlude Background Geometry" in 2.48. It is the cube button that appears toward the right of the 3D View header [note on edit: this is also not available in the depicted 'Draw Type -> wireframe' mode for later versions])

Press **CTRL+EKEY** and select *Mark Seam*, or select *Mesh* menu → *Edges* → *Mark Seam*. This tells the UV unwrapper to cut the mesh along these edges.

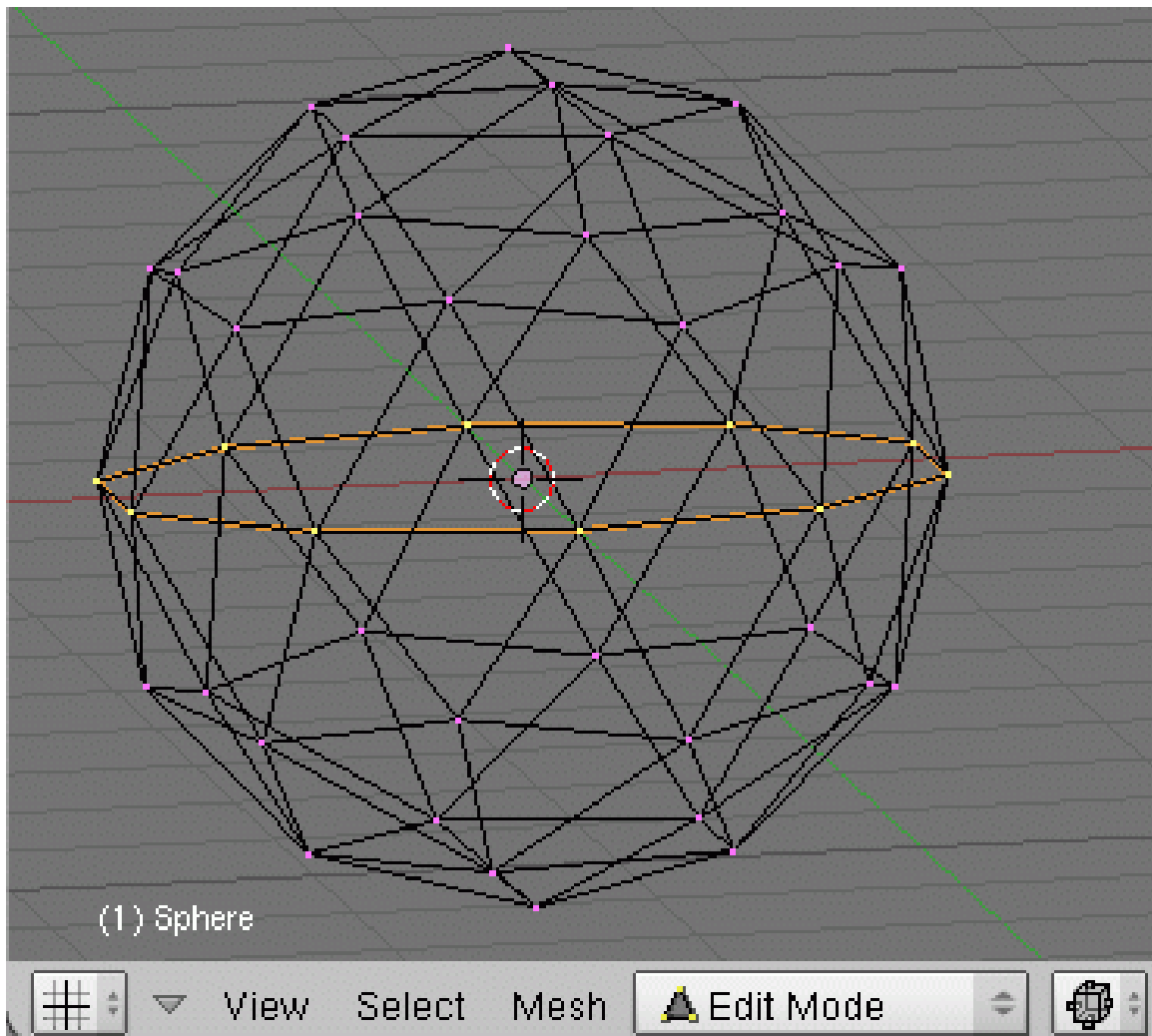


Figure 677

Unwrap the mesh

Blender 2.49 is a complete revision, this method wont work but contains useful information and notes by users. For a finishing method that is current see 2.49b below².

Next, create a window for the UV mapping: click the **MMB** (**RMB** works as well) near the top border of the 3D View window and select "Split Area". Set its window type to "UV/Image Editor" with the drop down box at the bottom left corner of the new window or with **SHIFT + F10**.

² Chapter 80.1 on page 720

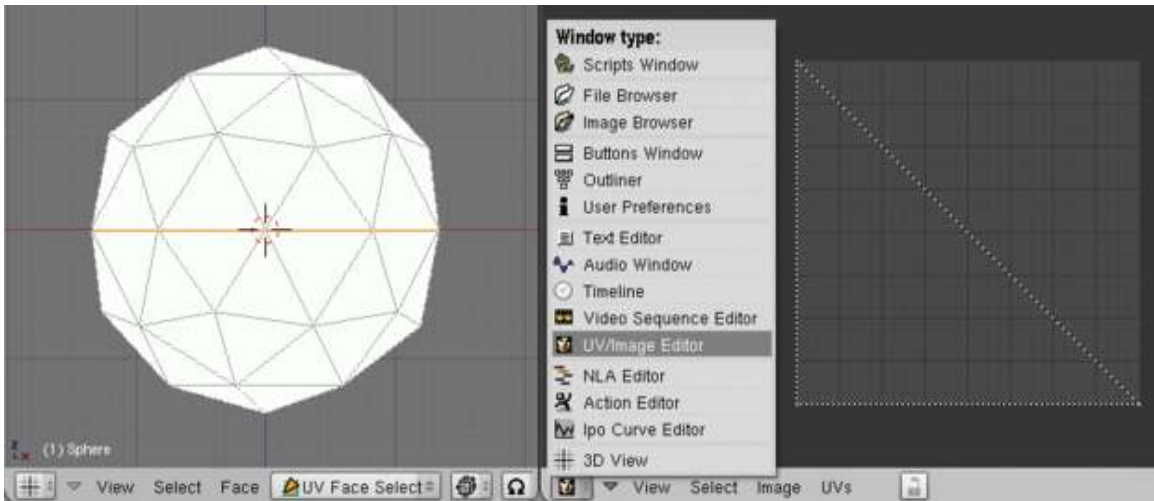


Figure 678

In the 3D View window, select all your vertices, and hit **U**KEY and then *LSCM*. For Blender 2.42 or later select all the faces, and use **U**KEY → *unwrap* or *Face* menu → *Unwrap UVs* → *Unwrap* to use LSCM.

Noob Note: I use Blender 2.45 and **U**KEY doesn't work for this. It seems to be an undo key. Is there another way to bring up the Unwrap menu? **Same Noob:** The tutorial doesn't say, but you have to be in UV Face Select mode for **U**KEY to work for this. **Yoshi:** For Blender 2.48a there is no LSCM but you can use Unwrap. **Thespeedylife:** for v2.48 and v2.49 the **U**KEY works in Edit Mode, and there is no UV Face Select Mode in these 2 versions.

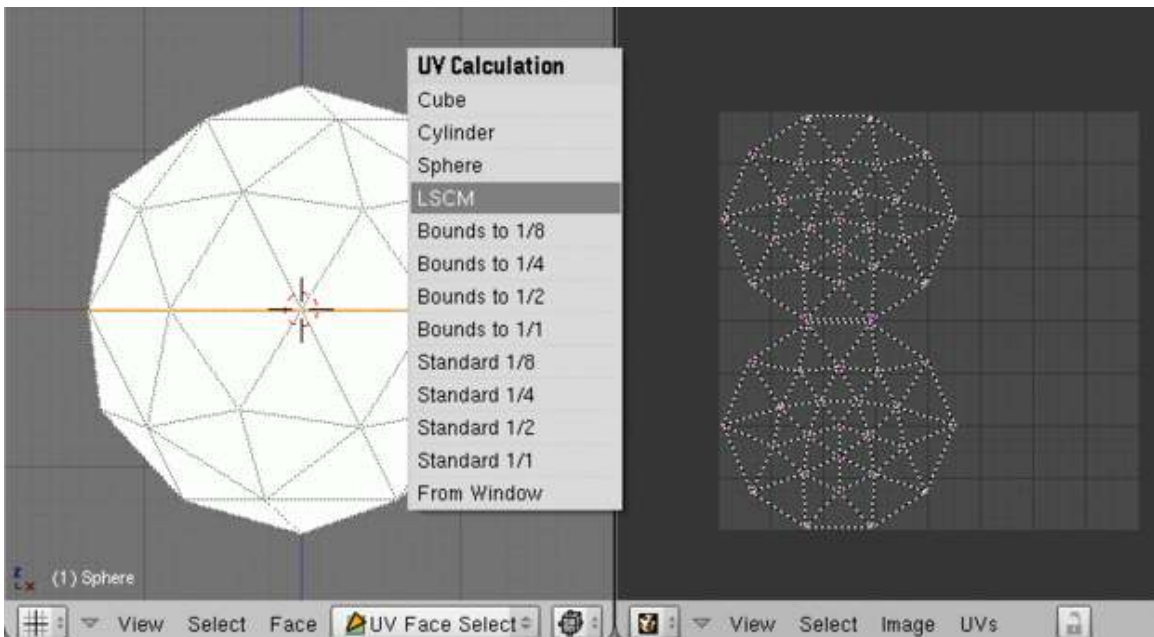


Figure 679

NOTE: Make sure you switch to the UV Face Select Mode in 3D window. Then "UKEY" does not work as a shortcut for Undo, rather it serves the purpose.

LSCM is one of the algorithms for unwrapping a mesh onto the 2-dimensional UV space, acronym for *Least Squares Conforming Map*. It is a very useful unwrapping method because it attempts to preserve the shape of each face, much like unwrapping the cloth of a garment.

Make a template image

When you have tweaked a nice layout and intend to make the texture image yourself, you may ease the texture drawing by saving an image of the UV layout. This image can then be opened in your image editing program of choice to make a basis for the UV texture by showing where each surface goes. In the UV/Image Editor window, select *UVs* menu → *Scripts* → *Save UV Face Layout...* (User note: for pre 2.43 versions, use *UVs* menu → *Save UV Face Layout*). <<**Noob note:** I use Blender 2.48. *UVs* menu → *Scripts* is empty. There's no *Save UV Face Layout...* option... What to do?>>**Yoshi:** I also use 2.48, make sure all the vertices are selected then hit *UVs* → *Scripts* → *Save UV Face Layout*. **New Blender User:** This is because you don't have the proper Python software installed. It is this software that provides all the scripts for Blender, so it is important to have the proper software. Note: Blender 2.48 requires Python 2.5.2, which is NOT the newest version. Python 2.6, the newest version, doesn't work with Blender, so make sure to get Python 2.5.2 (2.5.4 that is available at the Python site³ works too). There is another way to export the images, however, that does not require scripts. Go to *Image* → *New*, and this will create an image of the net. Then go to *Image* → *Save As*, and save the image to where you want it.

With *Wrap* selected, the layout will maintain its proportions, which is best for general purposes. Not selecting it will scale your layout into square proportions which is mostly useful for Blender's game engine⁴ where textures should preferably be square. Choose the location of the image file as desired (the default name is the name of the object to be textured) and press *Export*. *Tip:* if your image painting program supports layers, try putting the UV layout in a locked, transparent layer above the actual painting. If you do not alter the dimensions of the exported UV image in any way it will fit perfectly with your UV layout when the image is loaded back into Blender.

Apply an image

Save the following image(Click to view in full high resolution (4,096 × 2,048 pixels)):

³ <http://www.python.org/download/releases/2.5.4/>

⁴ <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FThe%20Game%20Engine>



Figure 680

Load it in the UV/Image Editor window by clicking *Image* menu → *Open...* (or *Image* menu → *Load image* in older Blender versions). Then with the very basic operations, grab, rotate and scale, adjust the unwrapped mesh so that it fits nicely on top of the image.

[noob note: when i downloaded this image it did not load correctly, try copying it into paint if you have this problem -> or better click in the link and it's send you to the correct link] [another noob note: when I used the image (in 2.49a/OSX) , it deformed the mesh, I found using a resized image (405 px. square - the same image as above placed on a square black background) sorted this out]

2.49b

Blender 2.49 changed the way UV mapping works. Skip this section for versions of Blender prior to 2.49.

After **3D View**:

1. (CTRL+E) Mark Seam.
2. (AKEY) Select All, then change the window type from 3D View to **UV/Image Editor** [2.49 **Noob Note:** I found it useful to split the window leaving one half as a **3D Window** and converting the other half to **UV/Image Editor**.]
3. Open: BlueMarble-2001-2002.jpg.
4. Click the 'belly button' (image selector) to select the picture. (You should see the picture. If it's black, try the mouse wheel to resize.)
5. Now unwrap. This will load the mesh horizontally on unwrap. [2.49 **Noob Note:** If you have split the window, as suggested above, the "U" key in the **3D Window** in Edit Mode will give a UV Calculation drop-down menu. It is also possible to click on "Mesh" in the headers of the **3D Window** in Edit Mode to get to the UV Calculation options.]
6. Then use the regular manipulation keys (RKEY, SKEY, GKEY) and buttons in **UV/Image Editor** to fit the uv meshes.

Observed: The uv mesh loaded alone is one vertical island while the uv mesh loaded on an image was two. Weird. In your Edit Panel buttons around now you can see the sub-context master panel called UV Calculation. This is like an important thing so don't worry if it is brain overload. Experimentation with that pack margin will separate the mesh islands in either case.



Figure 681

Admire your new creation

Back in the 3D View window, select Object mode. The next drop-down menu to the right is the Draw Type menu; use it to set the Draw Type to Textured. (Greybeard calls it "potato mode" because of the icon that the menu shows.) Hit TAB a couple of times to refresh your object, and admire your new picture mapped onto your object!

To make the texture visible in renderings, you also need to add the texture to the icosphere as a new material. In the Buttons window, switch to the Shading buttons by clicking the small shaded-sphere button or by pressing **F5**. Create a new material by pressing *Add New* in the *Links and Pipelines* mini-window, then turn on *TexFace* in the *Material* mini-window:

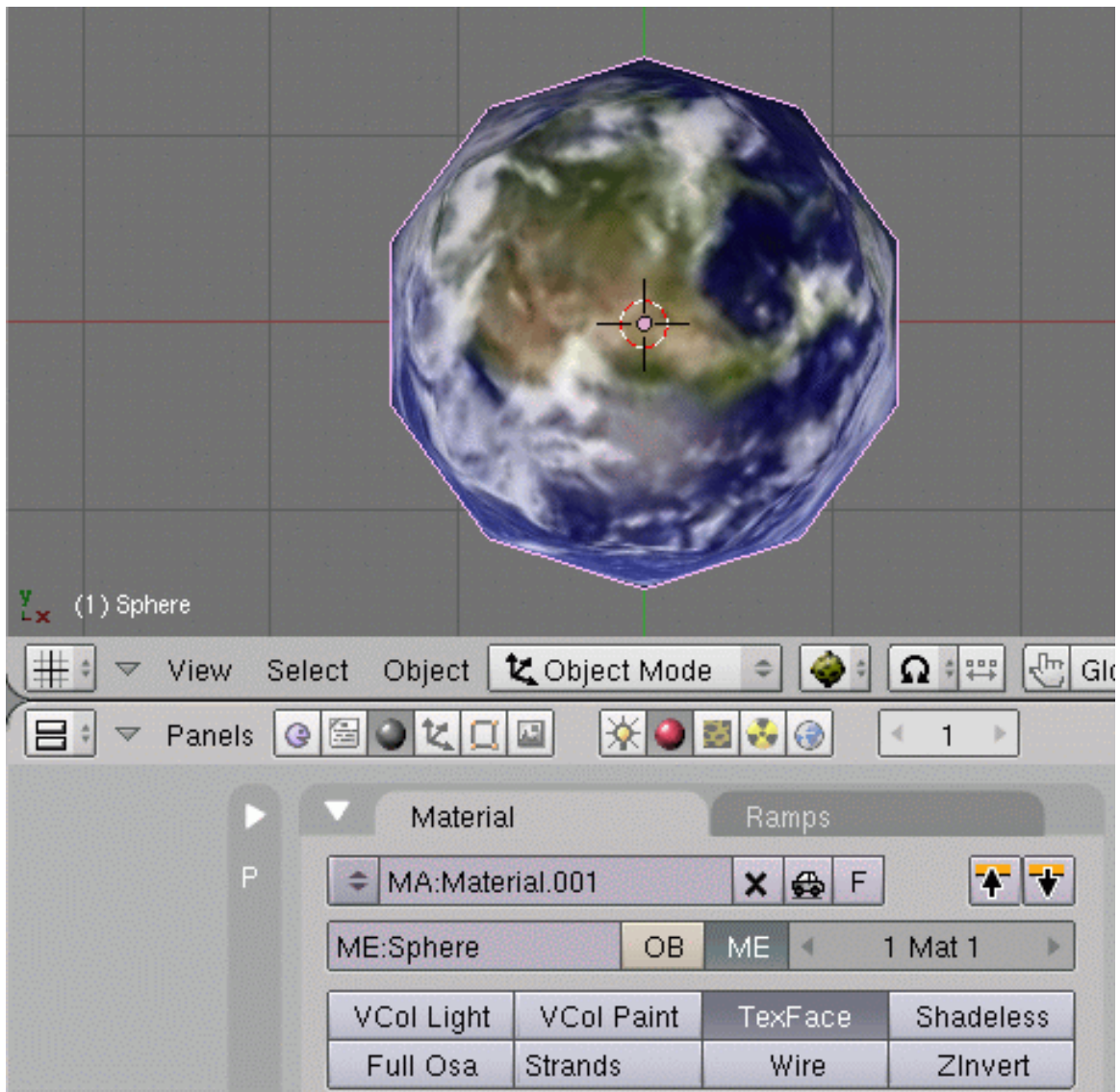


Figure 682

To finish your work, switch to Edit Mode and select all vertices. In the Buttons window, switch to the Editing buttons by clicking the small four-vertices-in-a-square button or by pressing **F9**, and then click *Set Smooth* in the *Links and Materials* mini-window. In the *Modifiers* mini-window, click *Add Modifier* → *Subsurf* and set *Render Levels* to 3. Switch back to the Shading buttons (**F5**) and activate the World buttons by clicking the small "Earth" button in the second button group. Enable *Stars* in the *Mist/Stars/Physics* mini-window. The scene is ready! Render it by selecting *Render* menu → *Render Current Frame*.

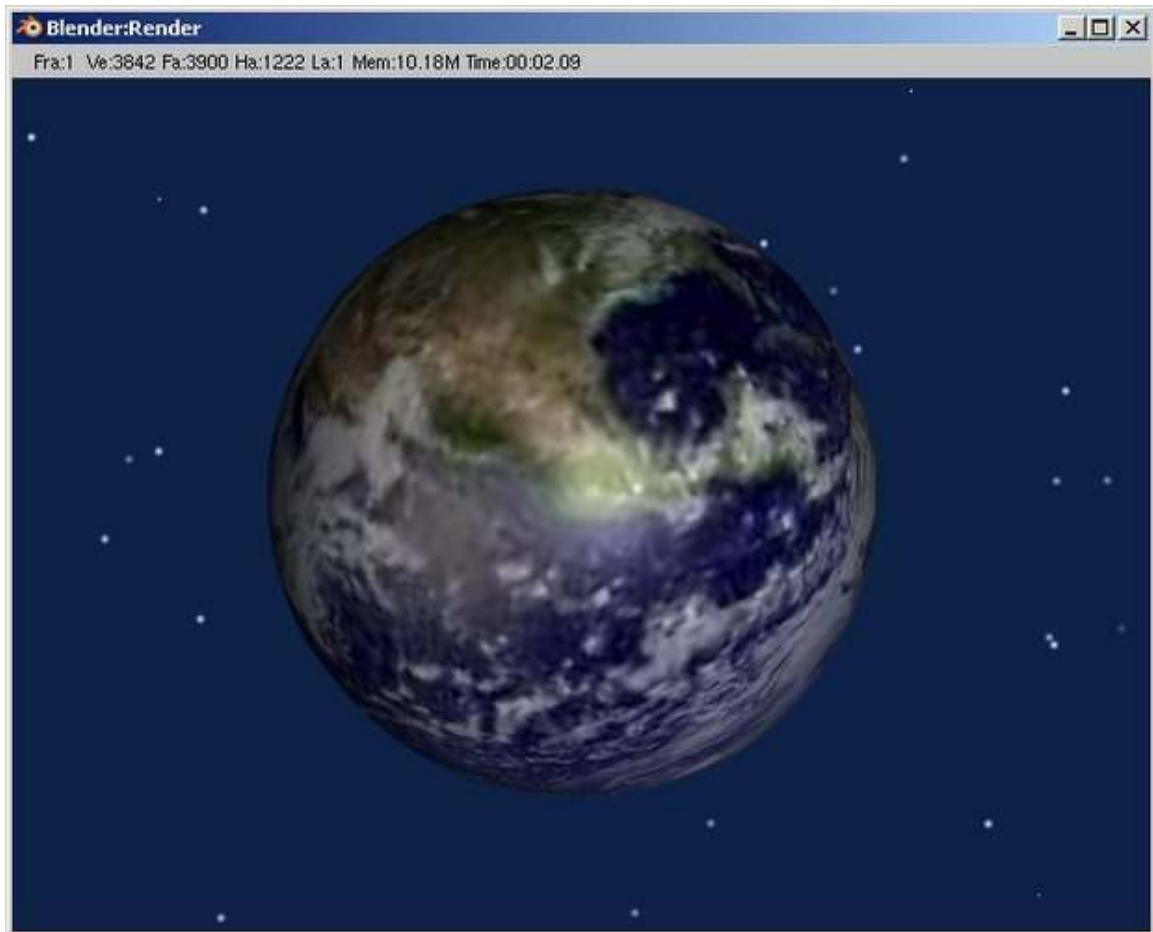


Figure 683

Some notes

All this relates to the UV/Image window.

If you are going to edit the layout of your unwrapping (so you can make a better picture), make sure Select->Stick Local UVs to Mesh Vertex is on. You can "pin" vertices when they have been unwrapped with **PKEY**. If you do, make sure you put at least one pin on every island. You can remove all pins with alt-p. LSCM works by trying to maintain the angles between each vertex. If an unwrapping should be symmetrical and it's not, try putting one pin in the middle of the outside edge of the big side and one on the small side, and unwrapping again with **EKEY**.

IMPORTANT:

- Remember to set your map input with the UV and Flat buttons enabled.
- As said before, to make the texture visible in renderings, too, you also need to toggle on TexFace from the Material buttons.

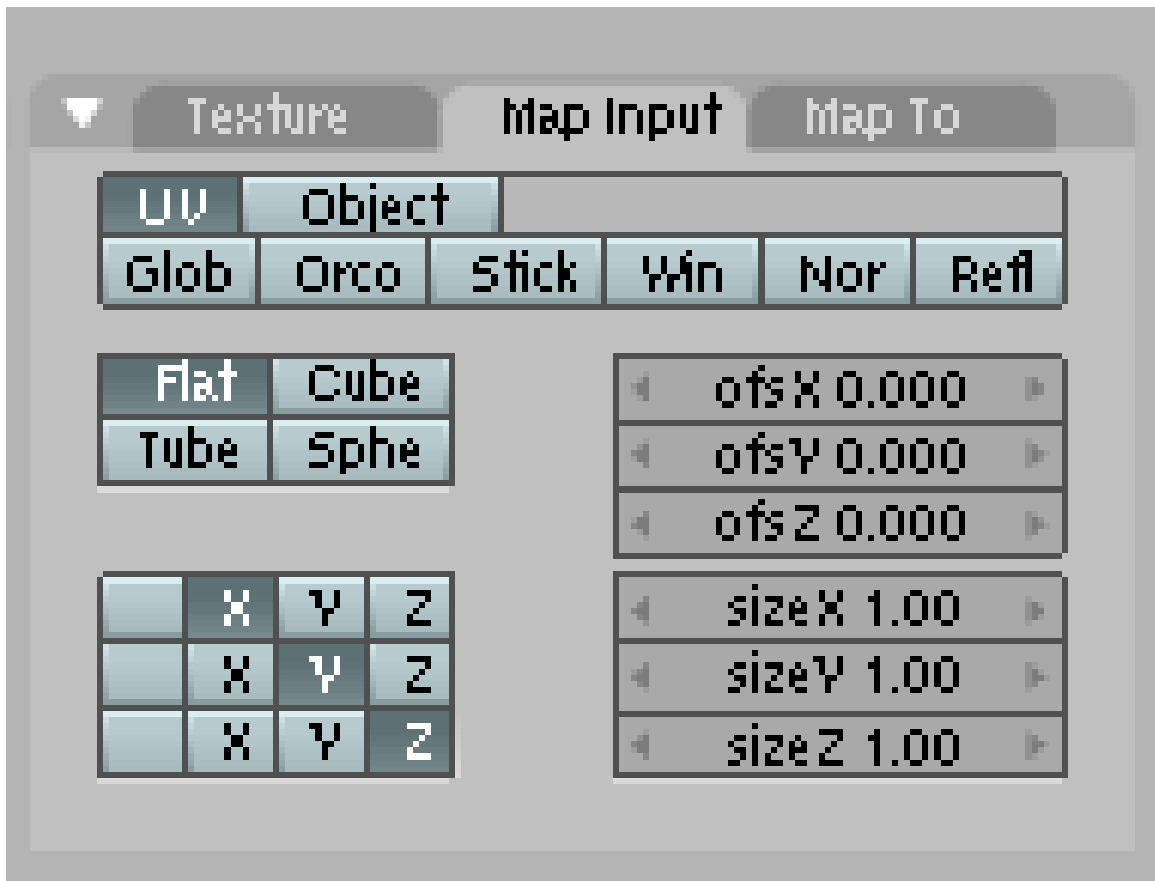


Figure 684

80.2. Questions

1. **Question:** I can't get the image to load in blender v.2.44, thought I did everything you said. What to do?
 - **Answer:** In v.2.44: click 'Image' button on the bottom of the UV/Image Editor view port. There is the 'Open' option for loading an image file.
 - **Answer 2:** There appears to be a bug in Blender. If you still don't see an image, try cropping your image to dimension that are power-of-2, like 256x256, or 512x512, or 1024x1024.
 - **Answer 3:** Make sure to select the image in the file browser with both the right AND the left mouse button
 - **Answer 4:** Undo (Ctrl + Z) one step also helps to refresh the window and makes the image visible again.
2. **Question:** Whenever I press U in the step above everything is deselected and nothing appears in the UV window. What to do?

- **Answer:** First you select the circle on the icosphere (the 'equator'), ctrl-E, mark seam. Then where it usually says Edit Mode or Object Mode, click that and select UV Face Select, THEN Press U.
 - **Elaboration:** I'm experiencing the same thing even if I hit ctrl+E and mark seam. And AFAIK on 2.48a there's no UV Face Select mode. Well after hitting **UKEY**->*Unwrap* there's no projection of the sphere on the UV window. *Update:* I found the problem. In the 3d window you have to select all vertices in edit mode. Then the projection will show up.
3. **Question:** My image won't load and is there a way to load images other than typing in its entire address? (I.E. is there a 'browse' button?)
 - **Answer:** While there is no visual list of common shortcuts, if you type in the crude location of your file and double click on directories to get to the fine one containing your file.
 4. **Comment:** I selected "Save target as.." instead of "Save picture as.." in my browser, so tried loading a saved HTML file instead of the jpeg. Also it seems you need to reload the directory listing in the "Open" dialog box in Blender else it won't find the newly saved file. You can do this by going to the parent directory and back into the child directory.
 - **Response:** Normally the image file used by the html file will be downloaded along with the html file. Loading an html file as your texture is not recommended.
 5. **Comment** There does not seem to be a "UV Face Select" mode in blender 2.46 - There is a face select mode in 2.46+, but, first go to edit mode, than find the small triangle icon in your window's toolbar, hovering over it will give you the tooltip: "Face Select Mode". you can also use the shortcut CTRL+TAB+3KEY in edit mode. (The default mode is vertex mode, the four dots icon)
 6. **Question:** Because 2.46 has no UV face select mode, how do I apply something like "two-sided" to a whole group of faces instead of just one at a time? - See the tip above on how to get in face-select mode
 - **Response:** Changing a material property for any user (even a single selected vertex) will change that for all the users of the material. You can assign faces their own material (those 3 vertex if the face is a polygon) even though they are normally selected with a mesh. With using Similar to Selection -> Material you can then perform other actions on those faces. Beyond that you can create groups that do not use materials as their commonality. I am a Noob, so I have never seen UV Select as a mode, but you can work out details before mapping, and there are some buttons like Sync which seems relevant.
 7. **Question:** when i press u it does not show the menu you have on your screen shot it has unwrap and some other things but does not say LCSM
 - **Response:** "LSCM" has been renamed to just "Unwrap"
 8. **Question:** When I finished the mapping, I had the normal earth sphere, but with this black ring where the seam was made. It's not a flat line.
 - **Response:** Make sure that all the verticies are inside the circumference of the Earth image. Also notice that each half of the Earth image is shadowed on the right side. Try rotating one half of the verticies 180 degress around so that the shadows on the Earth match up on both halves.
 9. **Question:** My earth won't show up in renders. I've mapped it, I've tried all sorts of lighting, but it won't show up. What should I do?
 - **Response:** You have to make sure (textface-in earlier versions) or (face textures- under materials tab (blender 2.58)) is selected. Then when you render, the uv map will show on your object.

10. **Question:**In the UV Image Editor panel, I can click on Image > Open and select the edited TGA file, but when I click Open Image, nothing happens. Is there a problem between Blender 2.49 and Python 2.6?
- **Response:**Nothing actually changes in the UV Image Editor window. You might need to change your Draw Type in a 3D window to Textured.
11. **Question:**UV export is not working: I have python 2.5.4 with blender 2.49b. I can generate the UV in Blender but I cannot see any of the script options (menu empty) even when all the meshes are selected. Furthermore when I try to export an image with save or save as, it just creates an empty tga with only the background color and no edges whatsoever.
- **Response:**Make sure blender is properly installed, in my case the installer forgot the .blender and plugin directory. Try the .zip file from the blender website instead of the installer. uv_export.py should be in the .blender/scripts folder. In the uv image editor use UVs -> scripts -> save UV face layout to export to tga.

80.3. Alternative method

Add a UVSphere: *Object mode, add, mesh, UVSphere, OKbutton,*

Add image in UV Edit mode: *image, open, ...* (select the 360x180 degree format picture)

Unwrap using sphere from view: *Edit mode*(make sure all vertexes are selected), *mesh, UV Unwrap, sphere from view,*

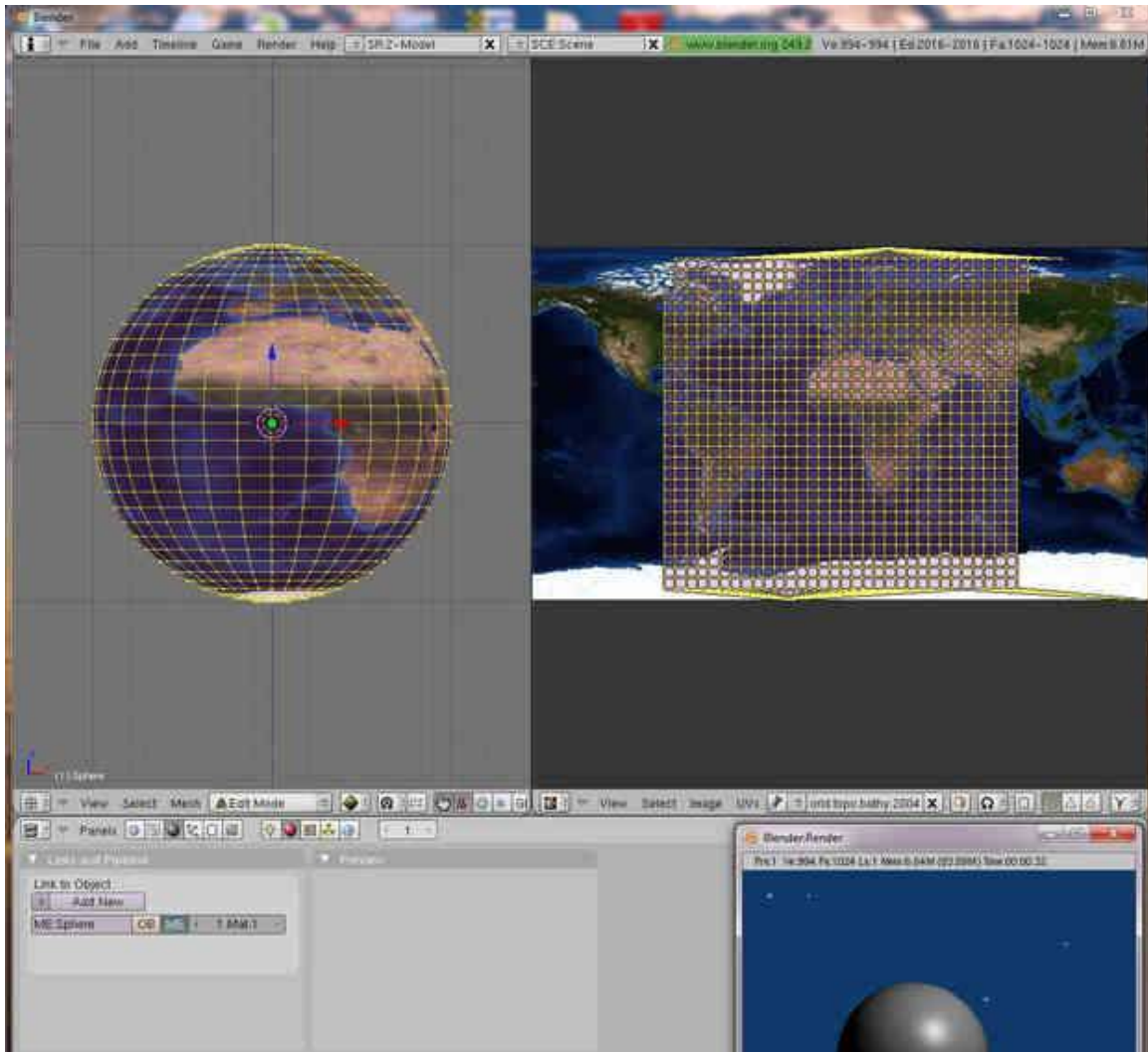


Figure 685

Adjust in UV Edit window the UV projection: SKEY, XKEY, GKEY, XKEY

Enable texture from UV Image: *Shading(F5KEY), Links and pipeline, Add new, Material, Texface*
 (Use UV image as texture)

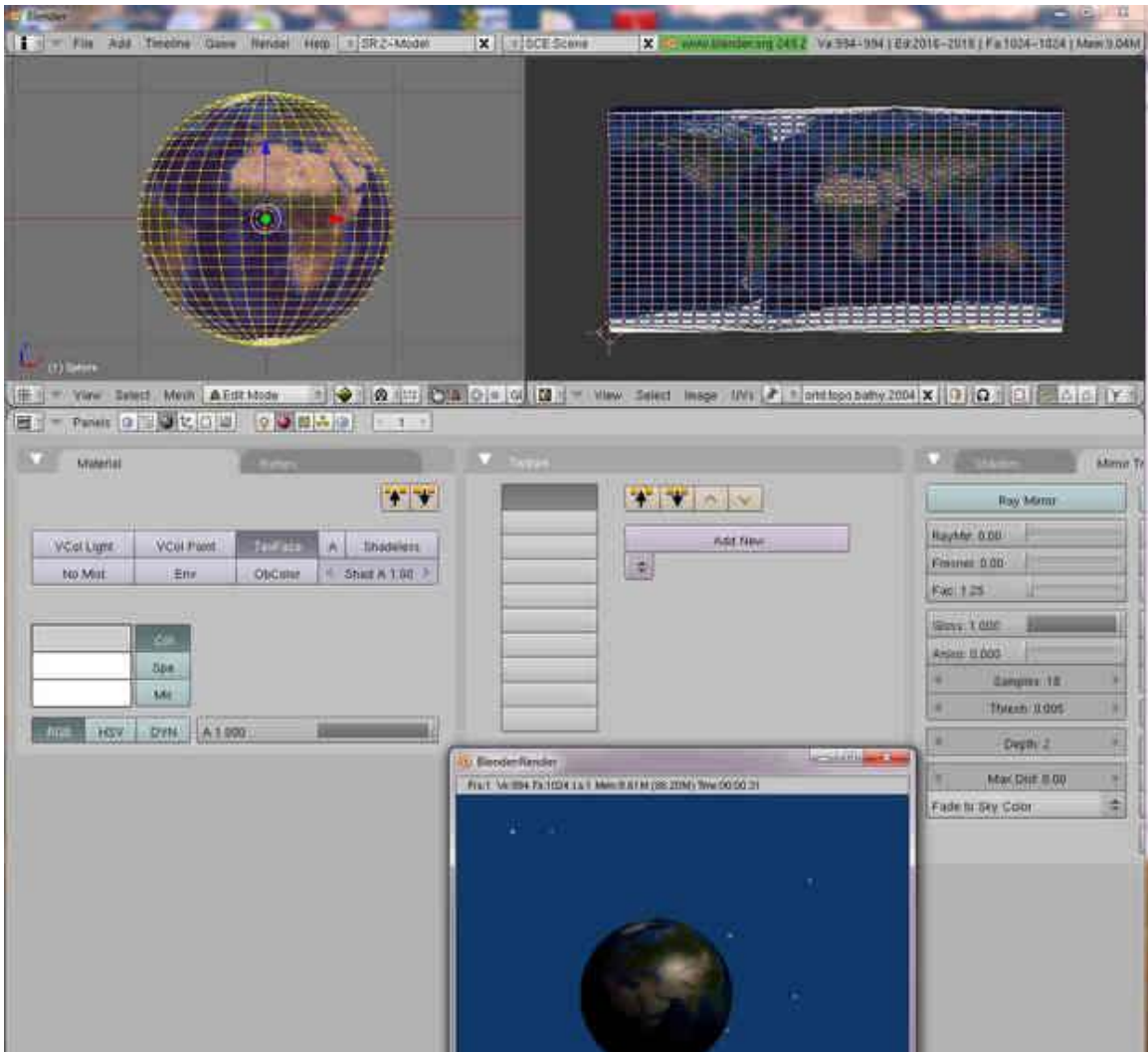


Figure 686

80.4. Source

Greybeard's LSCM Mapping video tutorial: <http://www.ibiblio.org/bvidtute/mytut/uvtut.avi> **Reference:**

1. http://wiki.blender.org/index.php/Doc:Manual/Textures/UV/Unwrapping_a_Mesh

This is a good reference page for the functions that blender provides. You can read additional pages of the manual if you need extra clarifications.

1. http://earthobservatory.nasa.gov/Features/BlueMarble/BlueMarble_monthlies.php

Sources of detailed world maps including surface heights(topography) and ocean depths(bathymetry) in the 360x180 degree format.

80.5. Texface versus Map input - pasting a photo to a sphere

Add a UVSphere:

object mode, add, mesh, UVsphere, shading, material buttons, Add new

1. Using UV Map editor, i.e. enable *texface* in *shading, material buttons*:

Select 3KEY(side view), RMB(select object) ,*edit mode, mesh, UV Unwrap, Project from view, SKEY(scale), GKEY(move),... shading, material buttons, texface*

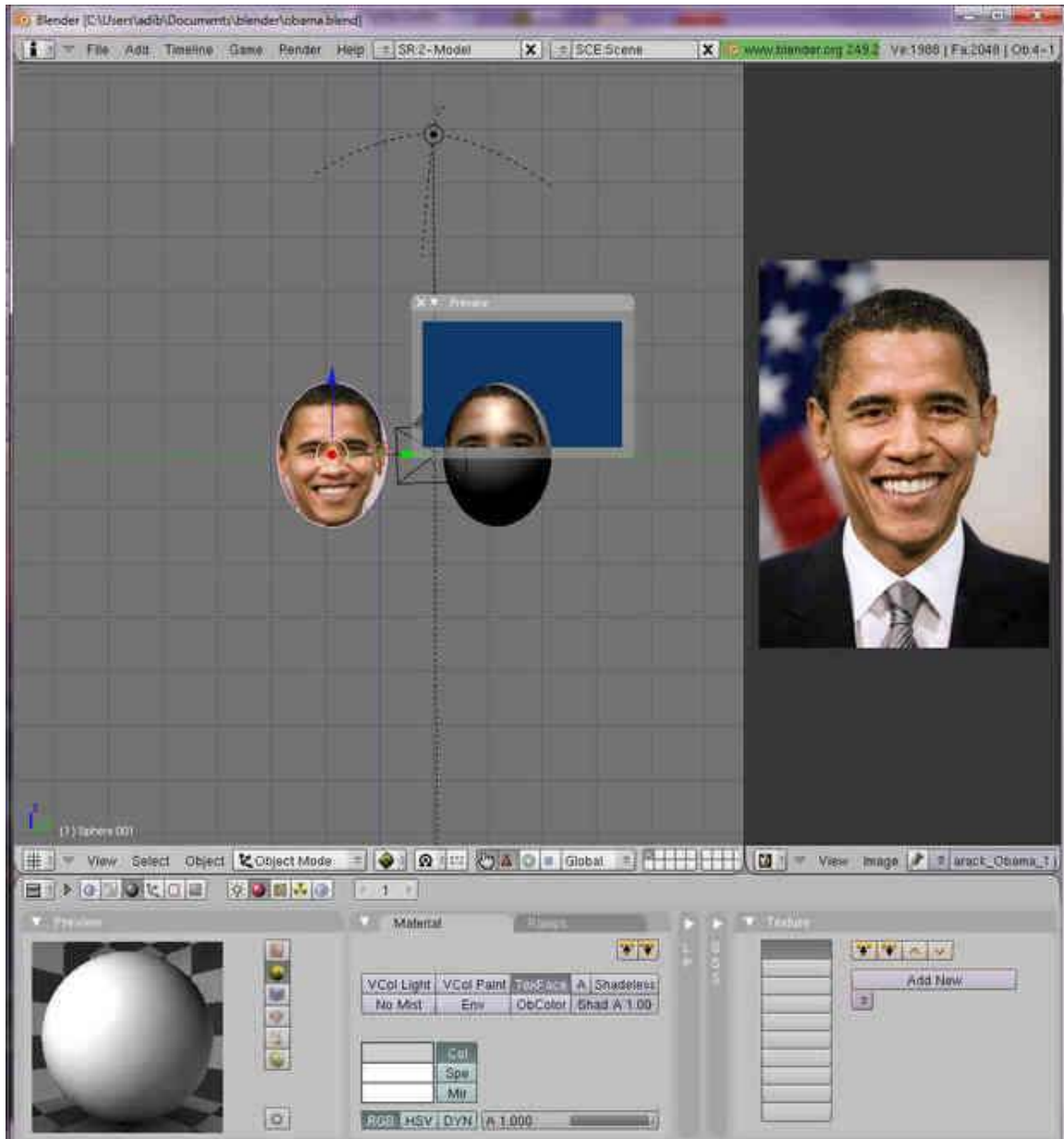


Figure 687

1. Using normal adding of image texture but with adjustments done at the *shading, material buttons, map into*:

shading, material buttons, texture, Add new, Texture button(F6KEY), Texture type, Image, (select your image), material button, map input, orco(original coordinate), sphere(3D object used), ofsX(X position), sizeX(X size)

In the *map input* button, adjust the ofsX, ofsY and sizeX, sizeY parameters until your image matches with the sphere.

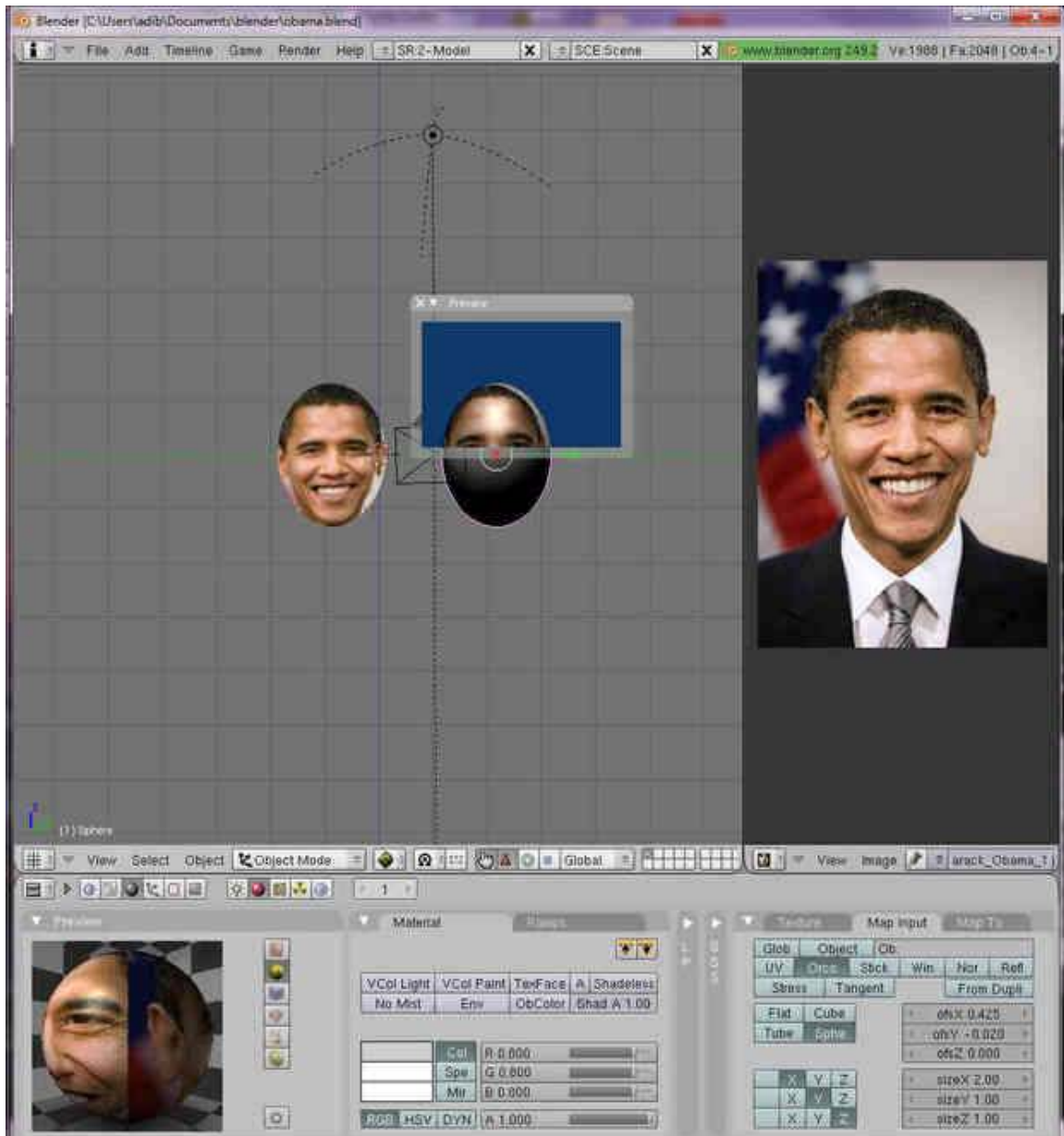


Figure 688

Notice the differences in the reviews that are available. The result may be the same, but the review options available are not the same.

Change the lamp into a hemisphere for a good lighting.

Reference: http://wiki.blender.org/index.php/Doc:Manual/Textures/Options/Map_Input

80.6. Video Tutorial

Ira Krakow's Blender 2.49 UV Map Basics Video tutorial: http://www.youtube.com/watch?v=I_-8OV92HLPY

81. Realistic Eyes In Blender

81.1. Overview

This tutorial will teach you how to effectively utilize textures and materials in Blender to create realistic eyes for characters. This tutorial was inspired by a great Maya tutorial in the Gnomon Workshop series by Alex Alvarez. My goal with this tutorial is to teach you how to get those same stunning results using Blender. Please check out Alex's tutorial as well, as it is very informative and covers more detail than I will be covering here. This tutorial assumes you know your way around the Blender interface and so I will not explain each key command as I go, but will instruct you in what steps to take and point out details such as key commands only when it seems relevant to do so. In addition, you will need some sort of image manipulation software, as this tutorial relies heavily on image-based textures, which we will be preparing outside of Blender. I will be using Photoshop, but feel free to use GIMP or whatever you're comfortable with.



Figure 689

(Note: image was removed by moderators because there is no WikiBooks license that allows me to post an original piece of artwork without giving up the rights to it. Images hosted on external sites cannot be embedded either.)

Above is an example of what can be achieved using the techniques described in this tutorial.

OK, let's dig in...

81.2. Reference Material

Let's Take a moment to familiarize ourselves with the anatomy of the human eye. Do a Google image search and examine whichever diagram is to your liking. Now, you don't have to pay attention to

every single feature, but do take note of the **cornea**, the **iris**, the **pupil**, and the **sclera**. These are the only parts that are visible to us when we look at a person's eyes, and therefore the only parts we will actually need to create. Many of the diagrams you find out there will be disproportionate, and so in order to know how large to make our features in relation to one another, it is best to view a large number of diagrams, a large number of eye photos, and also photos of plastic models of the eye, as plastic models tend to be truer to scale than 2D drawings. I also found a photo of a guy popping his eyes half-way out of his head. All of these together gave me a good feel for what the appropriate proportions are for a human eyeball. To some extent though, cornea size and sclera shape will vary on a person-to-person basis, so there is no precisely correct size, per se.

81.3. Building The Eye

Create a UV sphere. 24 segments, 24 rings, and a radius of 1 is recommended. The top of the UV sphere will become the cornea of the eye. So now let's go into **side view**, in the **orthographic** view, and with the topmost vertex selected and **proportional falloff** enabled, **grab** and raise this vertex along the **Z axis**. Don't click to finalize the edit just yet - first we will use the **mouse wheel** to control how much falloff is going to be applied. This process can require some finesse and multiple adjustments until the bump of the cornea is anatomically correct. You will want to set the falloff shape to "**sphere**".

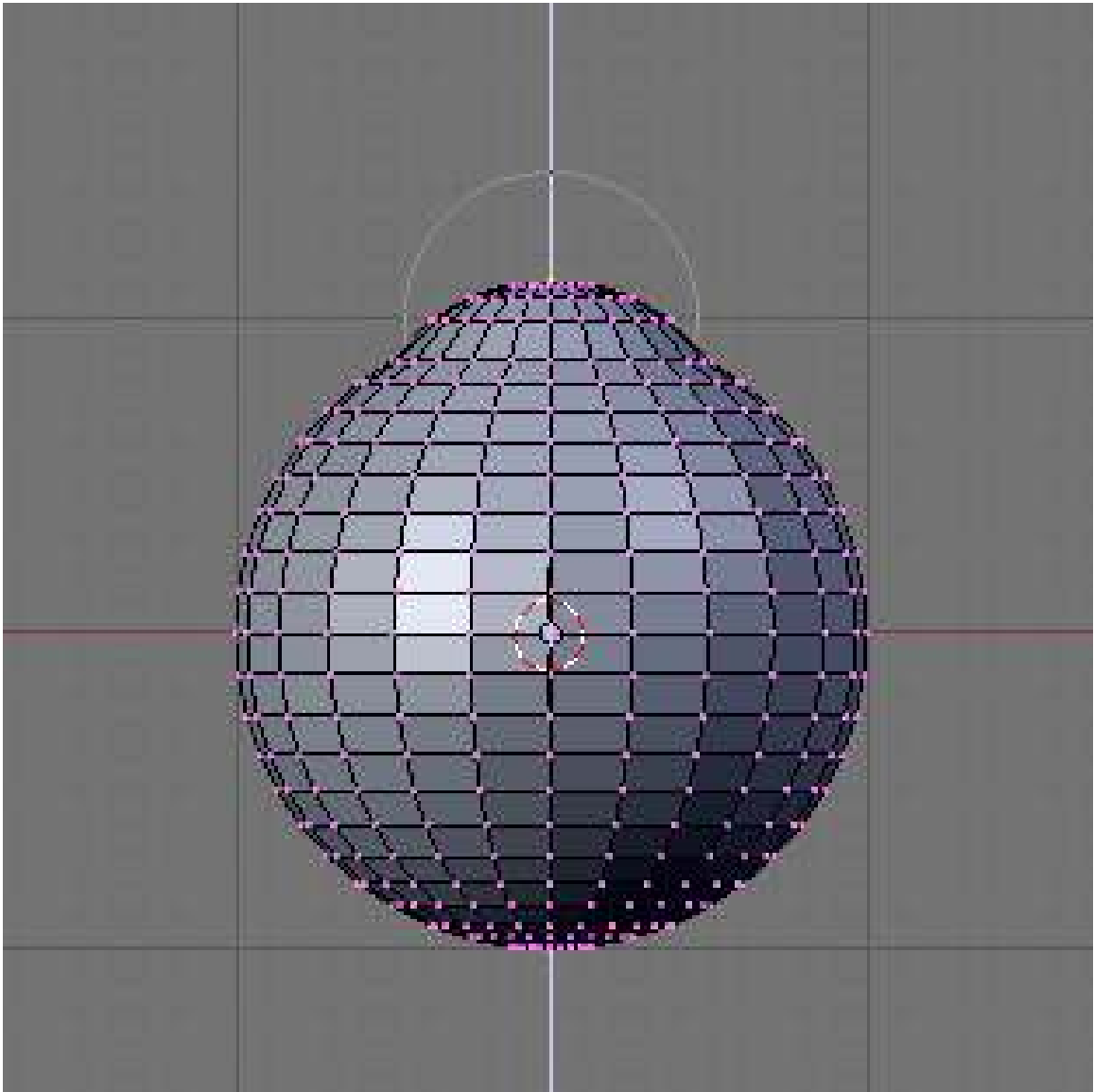


Figure 690

Below is a render of the curve I ended up going with. You can use it as a guide when modeling if it helps you. *This is with a SubSurf of 3 applied, so keep that in mind.*

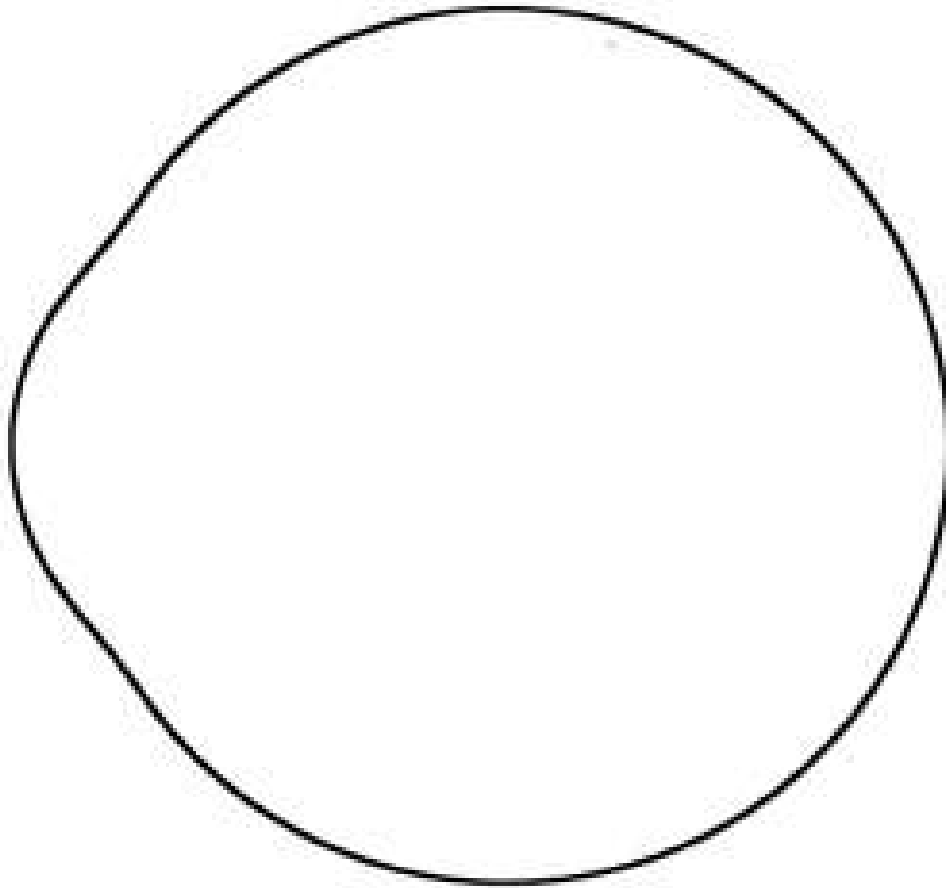


Figure 691

Now, some people may prefer to use a NURBS sphere for the geometry of their eyeball. If you are comfortable with NURBS modeling, and texturing NURBS models, go right ahead. It will give you a smoother and more rounded eyeball. I prefer not to complicate things with NURBS when polys will do just fine, and am guessing most of you feel the same, so I'm sticking with polys. This does however bring us to the topic of the SubSurface modifier and how it will affect your eyeball. When SubSurf is applied to a UV sphere such as our own, it causes the previously spherical shape to become slightly oblong. In other words, the eye will become nearsighted. This is barely noticeable, but it does happen, as a result of the way SubSurfing affects a UV sphere. You could create a UV sphere with more rings to ease this if it bothers you. It looks fine for our purposes though, to be honest. See the comparison in the animated GIF below.

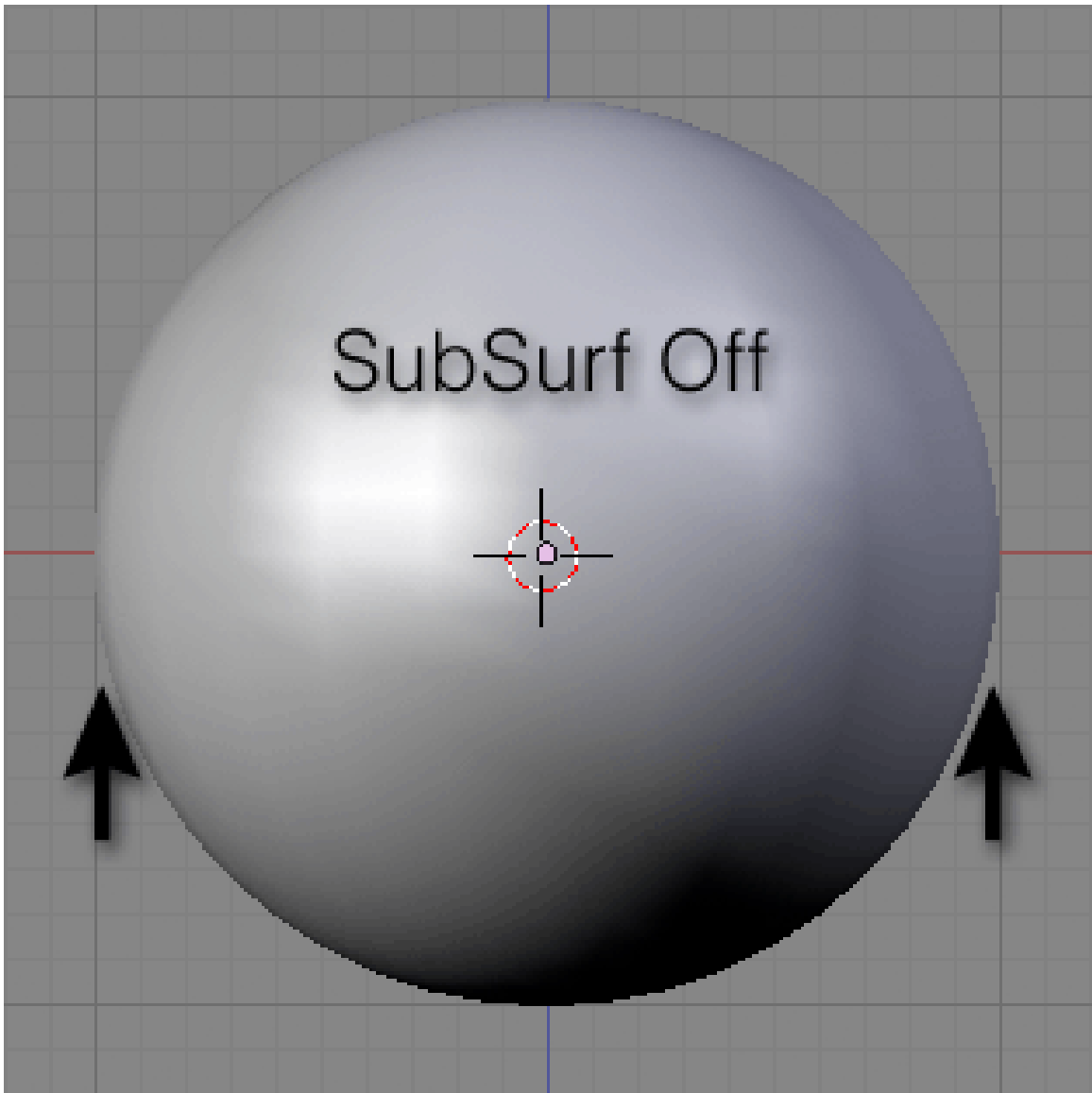


Figure 692

There is one last issue to address with regard to using a polygon-based eyeball. That is the small triangles that persist on the poles of the eyeball. These stubborn tris will remain and you will find that they result in an ugly and unnatural pinched effect when a specular highlight appears over the center of the cornea. Adding a SubSurface modifier will not make them go away. This didn't become apparent until after I did some texturing work and test renders, but I want to point it out to you now, before we get into all that. (See examples below)

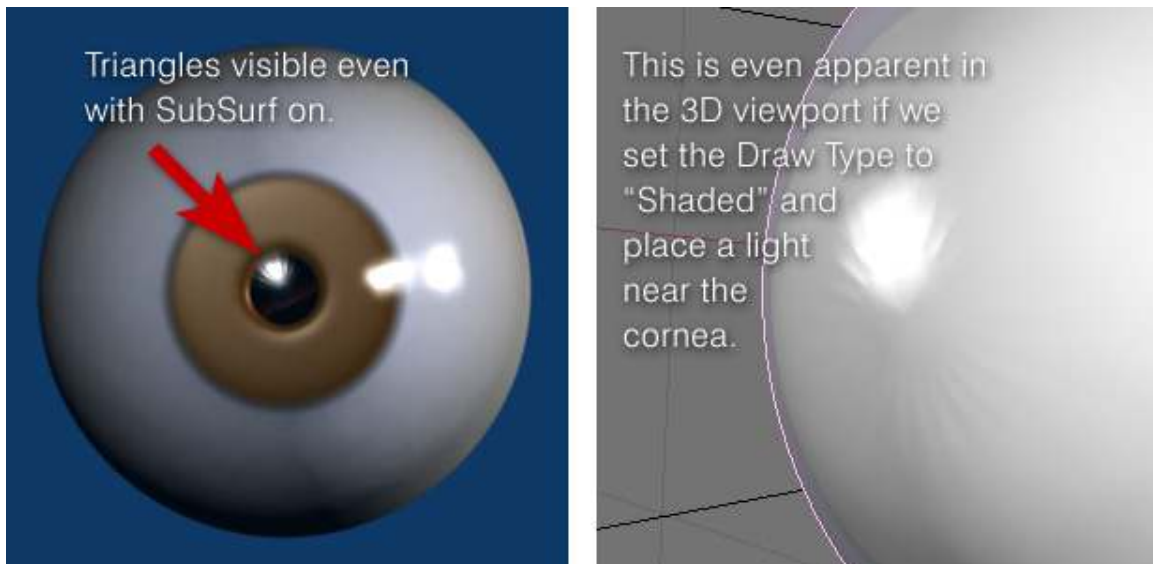


Figure 693

This can be remedied by adding more rings where those edges come together. Select only the edges in the very center of the cornea and **Subdivide Smooth**. Now **deselect** and again **select** the new innermost edges. **Subdivide Smooth** again. You can even do this a third time if you find it necessary. Rotate the model around in the viewport and do a couple of test renders to make sure the specular highlight doesn't reveal any artifacts.

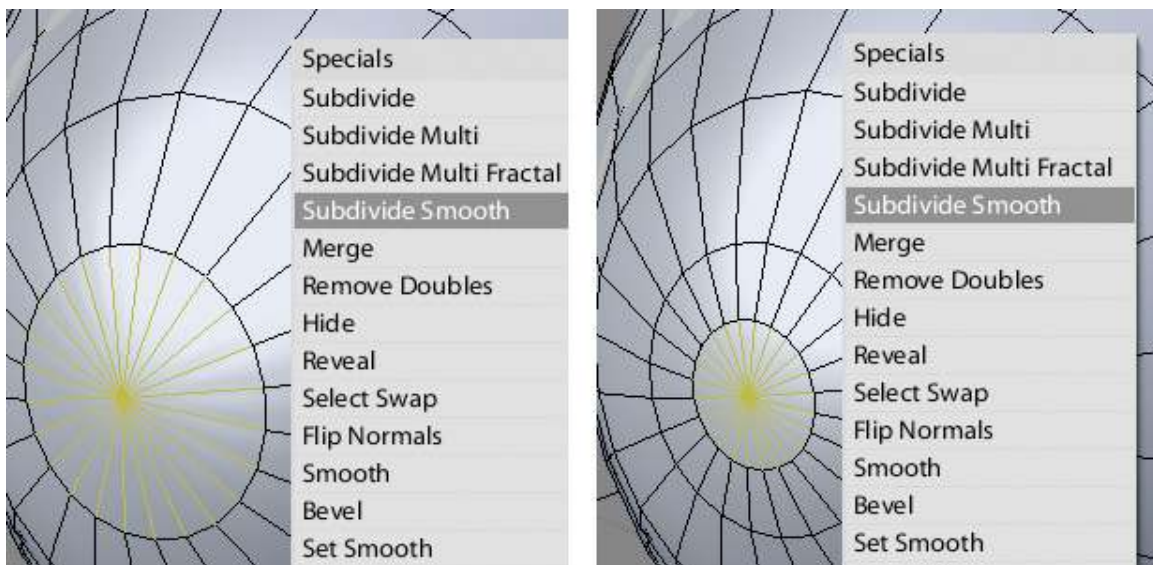


Figure 694

Now repeat this process at the opposite pole of the eye (the optic nerve area).

We are now ready to begin texturing the eyeball.

81.4. UV Mapping

In **Edit Mode**, we will define the seams for our eye. Select the edges you want to mark as seams and type **Ctrl E**, then choose **Mark Seam** from the menu.

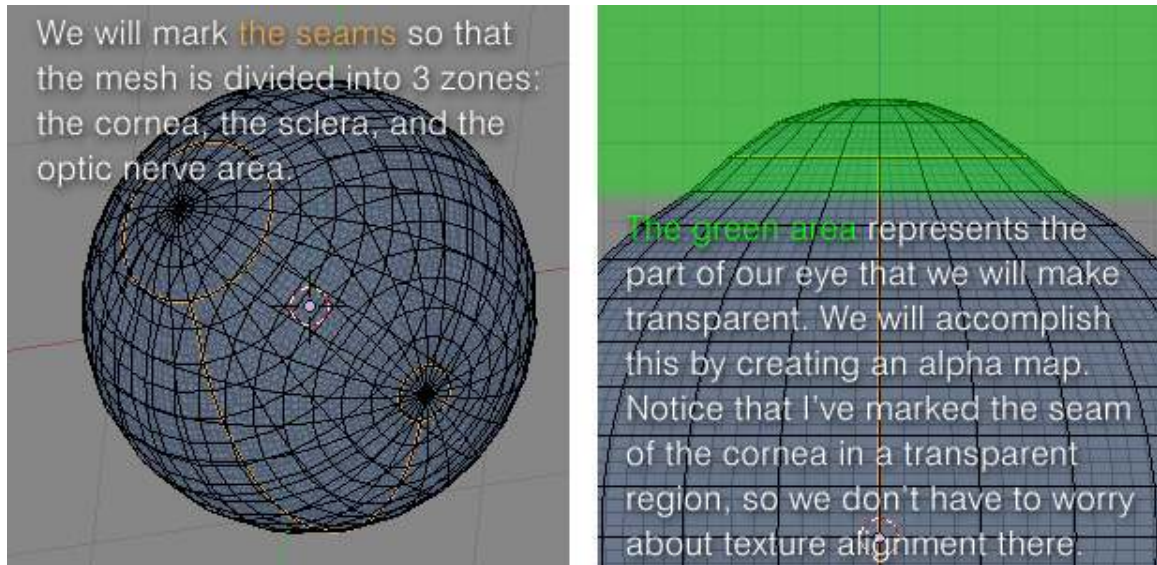


Figure 695

With all of your mesh selected, type **U** to unwrap. We can now see our unwrapped UV layout in the **UV/Image Editor** window. (For older versions of Blender, you might need to select **UV Face Select** mode to unwrap, but it's better if you download the latest.)

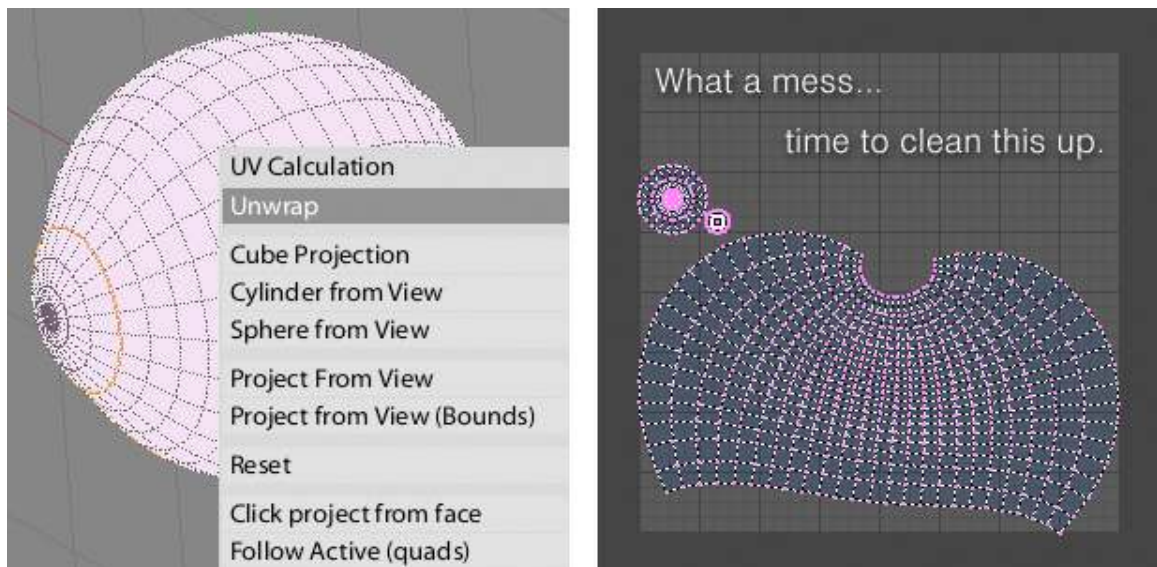


Figure 696

Those two islands we see are the cornea and optic nerve area. We can move them out of our way by selecting any vertex on them and typing **L** to select all linked vertices. Then **G** to grab as usual.

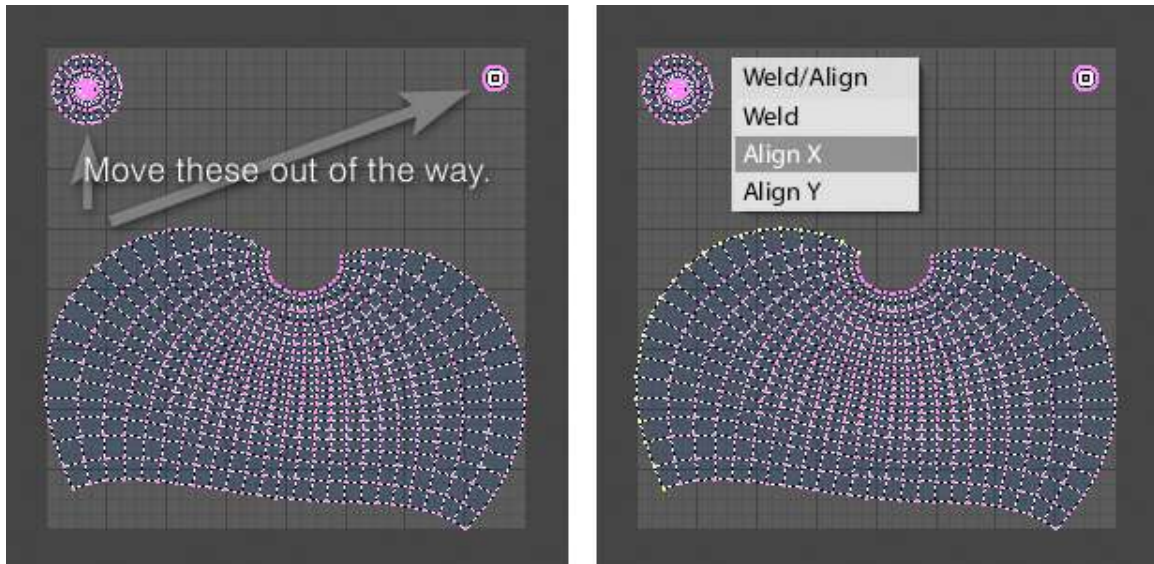


Figure 697

Select a column of verts on the sclera region of our UV layout. Type **W** to bring up the **Weld/Align** menu. Choose **Align X**. This is how we will go about straightening this contorted shape into a usable guide to map our textures.

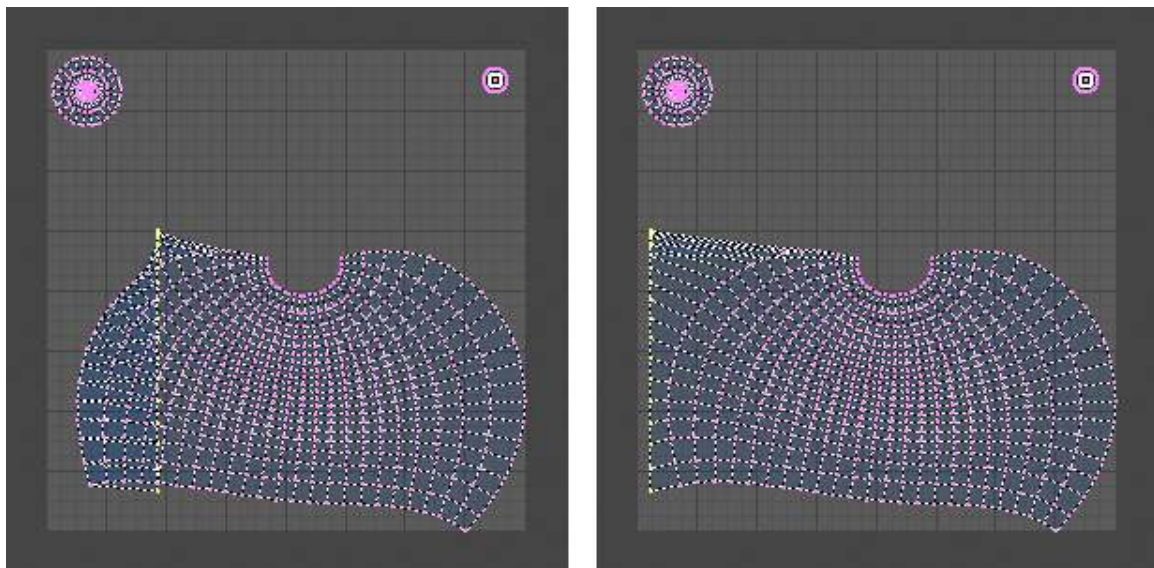
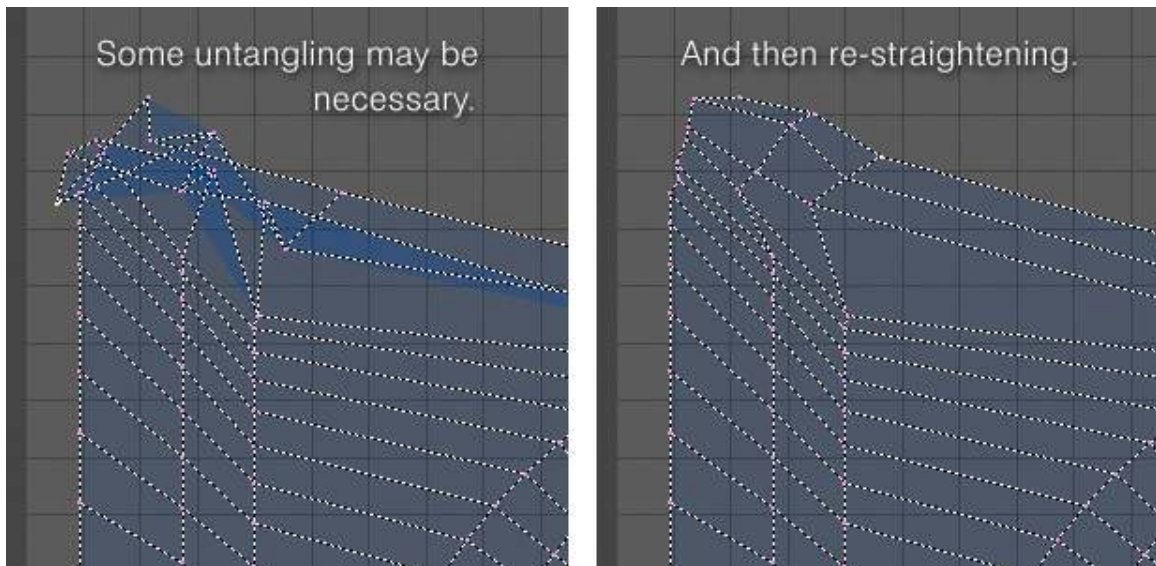


Figure 698

Grab and move your straightened column of verts over to the left.

Constraining to the X axis, by typing the "**X**" key after you grab, is a good habit to get into while straightening these out.

**Figure 699**

Straightening out all of these vertices into a usable grid shape will take you some time. It's one of the more tedious steps of this tutorial, but it's worth it in the end because having a rectangular grid will prove very helpful when we are texturing our eyeball. You'll want to use Blender's grid as a reference and just align the rows and columns to it exactly. Well, you can't get it *exactly* aligned, since there is no snap-to-grid feature, but you can eyeball it and everything will work out fine. The grid lines must be perfectly straight though, so always constrain to either the X or Y axis when moving anything. You can scale the whole thing up later, once each row and column is evenly spaced. Be sure to leave some space between your UV layout and the border of the layout area - don't "bleed" it to the edge or you will have texturing problems later.

[EDIT by nick_chang207]: You can align them perfectly; all you have to do is select all the rows and columns one by one, (you can use Alt + R) and then select the axis along which you want them to be aligned. For example, select the top column and then press 'S' to scale, then 'Y' to align it to the Y axis, and then '0' to align them. For the rows, do the same thing, only select the 'X' axis instead of the 'Y' axis this time. Its pretty easy this way.

[Edit by redset] If it's not clear, you can actually untangle this way too - simply Alt+RMB to select a row of edges, S Y 0 and Return (or left-click) to align them all, then G Y to move them up to a grid line. Repeat this for each row of edges, then work across the columns using the X key to align your scaling and movement to the X axis instead. You'll end up with a nice neat grid, and if you aligned them to gridlines you can just scale the entire island up to fill as much space as you need.

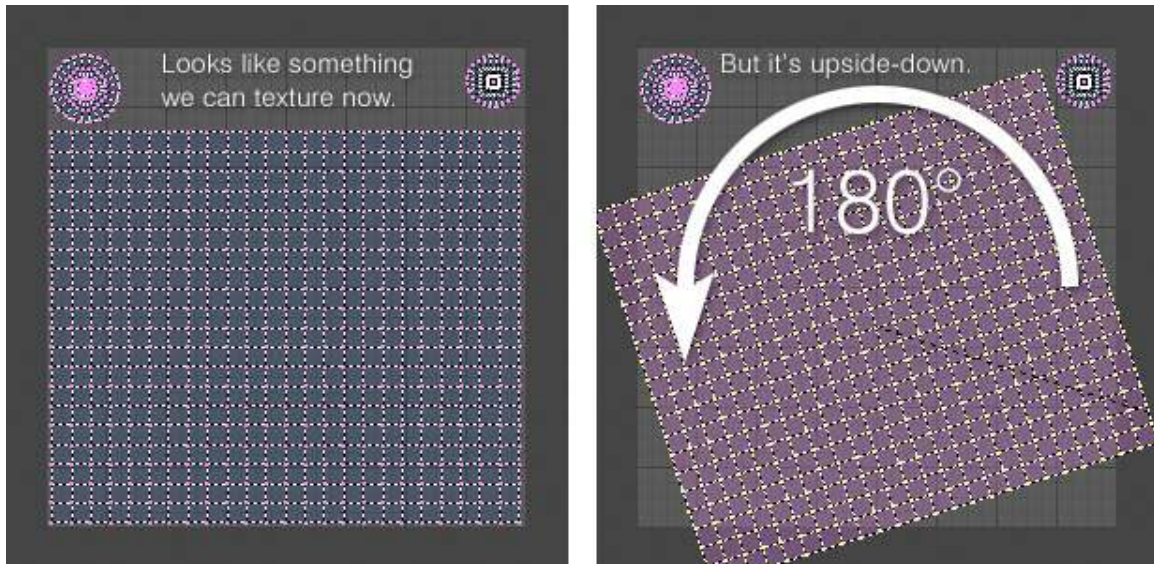


Figure 700

It's important to pay attention to is which way the UV layout is facing. Chances are the top of the sclera region is actually what you will want to be the bottom. It doesn't technically matter which way it's facing, but you will probably want it to look intuitive when you are painting the textures. I had to rotate that piece 180 degrees to get the cornea pole on top and the optic nerve pole on the bottom.

[Edit by redset] Mine actually seemed to be the right way up - I found you could check by going to the 3D view (or having split windows so you can see the 3D and UV views at the same time) and making sure every vertex is selected, then holding shift and clicking to deselect a vertex on the model, close to the optic nerve or the cornea. Back in the UV view the vertex will disappear from your UV map, so you can work out how the map corresponds to the actual mesh surface. There might be an easier way to check but I'm new to this myself! (Just select all again to get the whole map back)

Once you're done, **save your .blend file** and then **save your UV Face Layout**. This is done in the **Scripts Window: Scripts > UV > Save UV Face Layout**. The "UV Image Export" dialog box will appear. I recommend choosing a size of **2048** pixels.

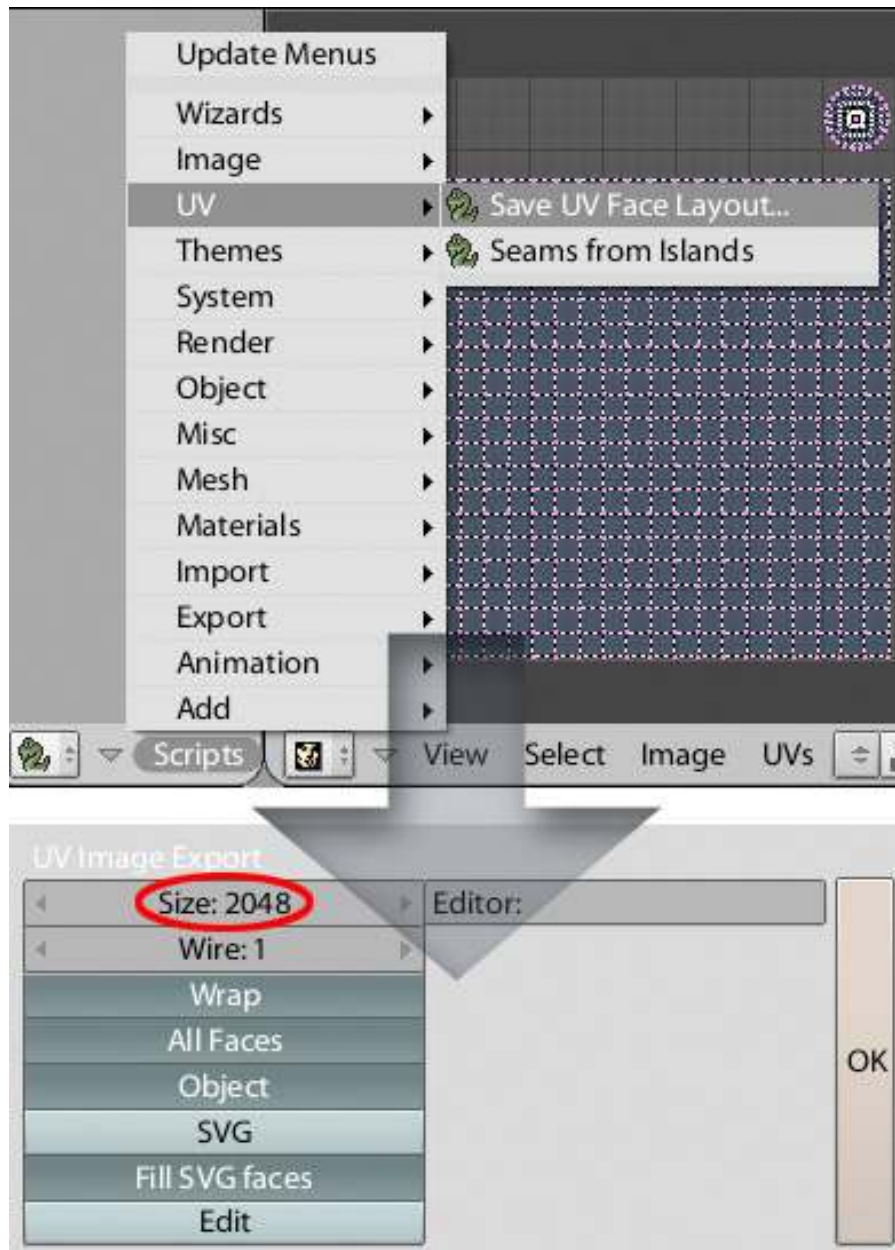


Figure 701

81.5. Preparing A Template For Texturing

Now launch your image editor and open the UV face layout you just saved. (You probably saved it as a .tga file, since that is Blender's default format for UV layouts) So we are now looking at a black and white UV grid. We want to get rid of the white regions so we can see through them to the layers below. The quickest way to do this is to set the blending mode to "multiply". If that works for you, go for it. Personally, I like my transparency to be represented by the checkered pattern, and for that we will need true alpha transparency. In Photoshop this can be done by going to the menu

and choosing **Select > Color Range**. Using the eyedropper, **click any white part** of the image, **set Fuzziness to zero**, and **click "OK"**.



Figure 702

Now **delete the selected white regions** by choosing **Edit > Clear** from the menu.

*nooby note: The same actions can be done in the GIMP editor by going to menu, choose **Select > By Color** and click any white part to select the white area. Then go to menu again and choose **Colors > Color to Alpha** and click **"OK"**.*

*another nooby note: In PhotoShop you can also modify the layer's blending options so that areas of white or black will become transparent: When you first open the image in photoshop it will be a single layer image, and the layer name will be 'background' - **Double click** it and give it a name like **"UV-Grid"**. THEN select the **"UV-Grid"** layer, **Right-Click** -> **"Blending Options"**, in the bottom area of the dialog set the top value for the slider on **"Blend If -> This Layer"** setting on grey to be a*

little lower (most likely 254). This means any values on "this layer" that are greater than 254 will be blended with the underlying layers. - Make sure you create your other layers **BELOW** this layer so the grid will always be on top.

One other element that will be useful to have in our template is a layer of markers that denote where the equator of the eye is, because what looks like the middle on the UV layout is not actually the middle. To find the middle, go back into Blender and count the rings of your eye mesh, starting at the seam of the cornea. Once you get to the equator, take note of how many rings it took to get there. Go back to your image editor and mark that area. You can mark it with a guide, or mark it with a coloring of some sort - as long as it's clear to you that it indicates the equator of the eye mesh. It may also be useful to mark the point where the transparency of the cornea should begin.

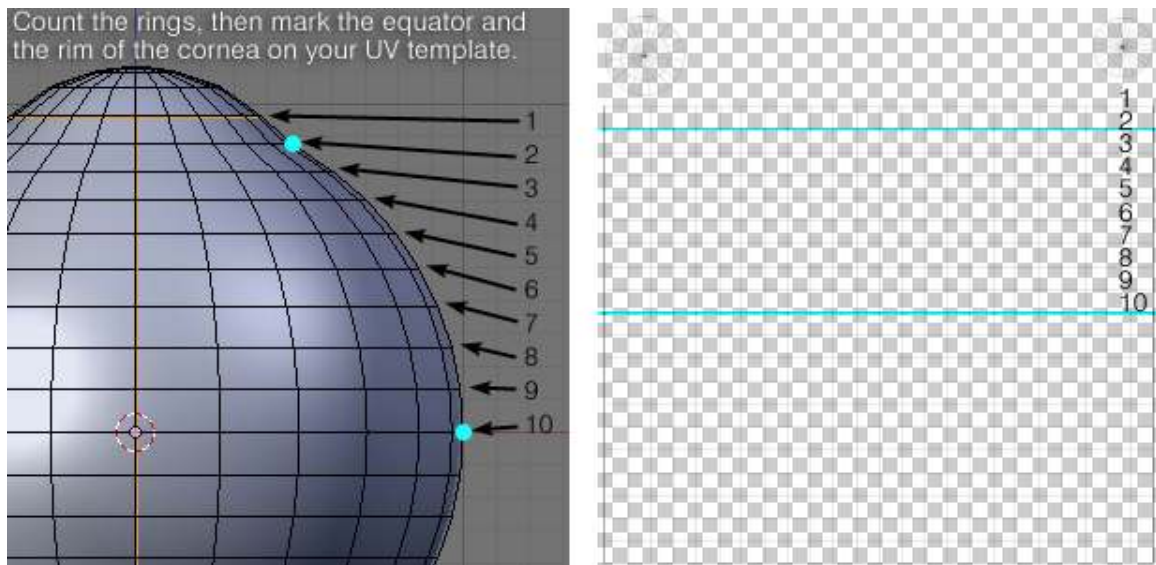


Figure 703

81.6. First Bump Map - The Sclera

Create a material for your eyeball. Then within that material **create a texture**. Name this texture something like "*BumpMap(Procedural)*". Set the texture type to **clouds**. A noise size of **0.25** with a noise depth of **2 or 3** is fine, and you can experiment with other values if you like. The other texture settings can be left at their defaults.

Now we are going to bake this procedural texture to get an image-based UV texture that we can edit in our paint program. Why? Because there is not an even level of bumpiness around the eyeball. The sclera has bumps, while the cornea doesn't. If your cornea was bumpy, your vision would be distorted. We want our 3D eyeball to accurately convey this subtle difference when light is reflected from the surface.

Before baking the texture, there are some parameters we must specify. In the materials panel click the **Map To** tab. The Color button is depressed by default, so you can leave it as is. Make sure this is the only depressed button. Then **change the color to black**.

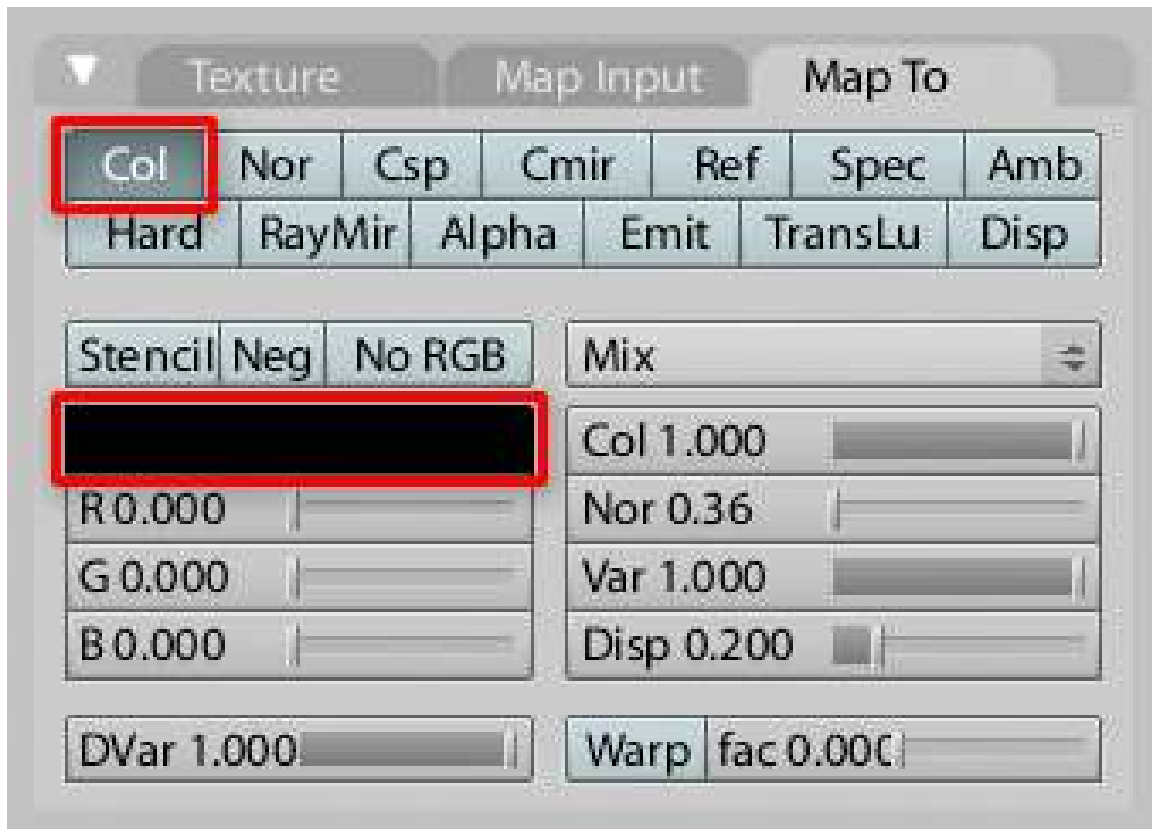


Figure 704

Now go to the **Material** tab. Change the diffuse color to **white** and click the **Shadeless** button.

Note: In this particular case it doesn't actually matter which of these colors is set to white and which is set to black. The end result is basically the same, and you can always invert the image later.

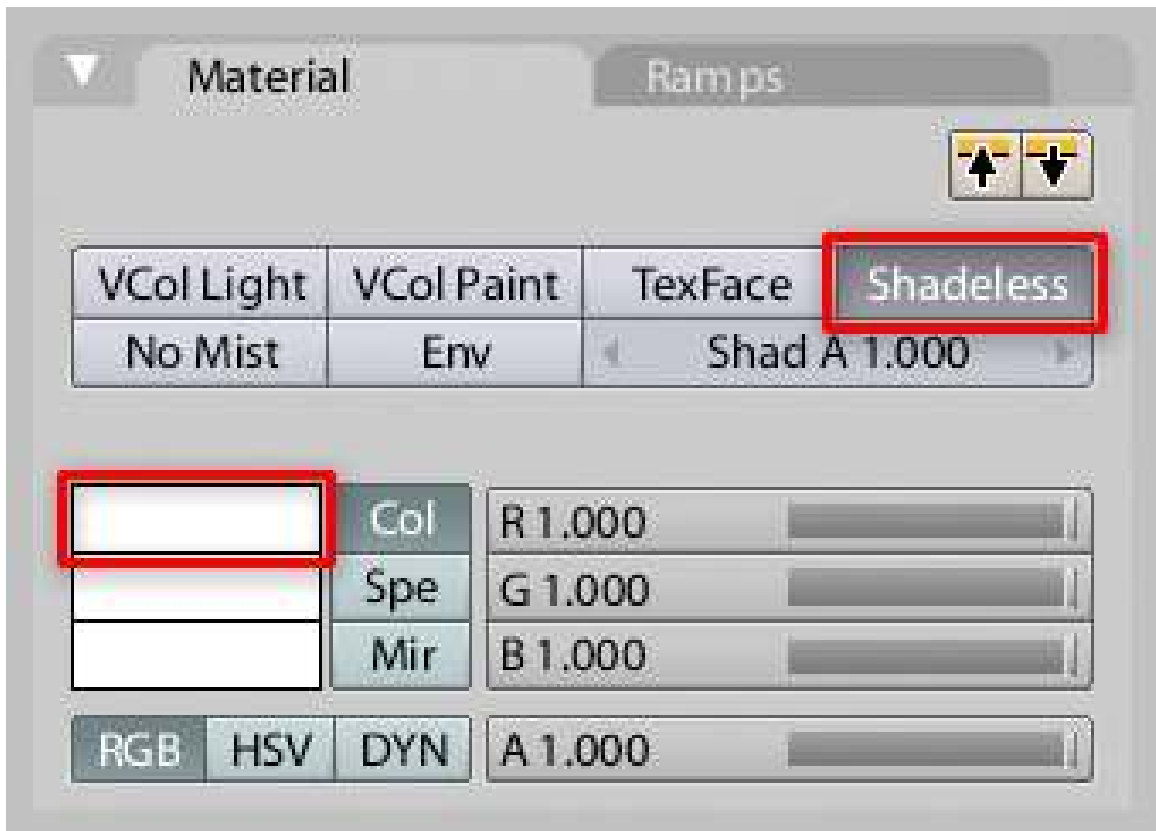


Figure 705

Check the Material Preview window to make sure it looks right.



Figure 706

Time to bake your eye. Make sure your eyeball object is selected and go into **UV Face Select Mode**. Then **split your screen** and set one viewport to the **UV/Image Editor**. From the menu at the bottom of the UV/Image Editor, choose **Image > New**. Enter a name like "*ScleraBumpBake*" and set both the Width and Height to **2048** (to match your UV guide). Click "OK" and a new all-black image will appear behind the UV grid.

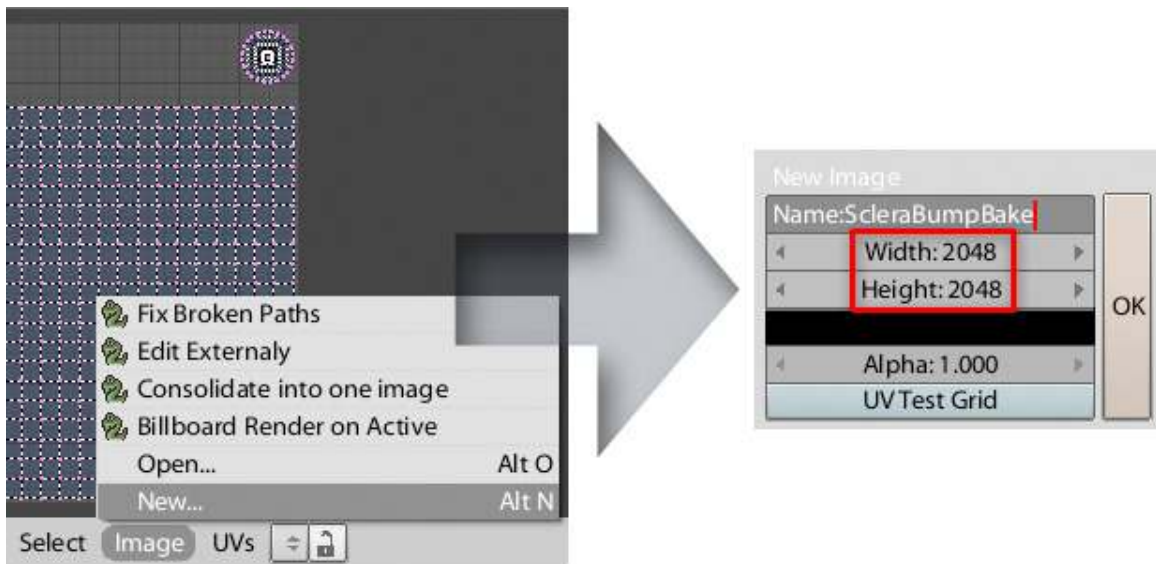


Figure 707

Go to **Scene panel > Render > Bake tab** and click the **Textures** button. Then set the margin to **10** and click the **BAKE** button. You will see the black image in the **UV/Image Editor** fill in with your bump texture. From the menu choose **Image > Save As**.

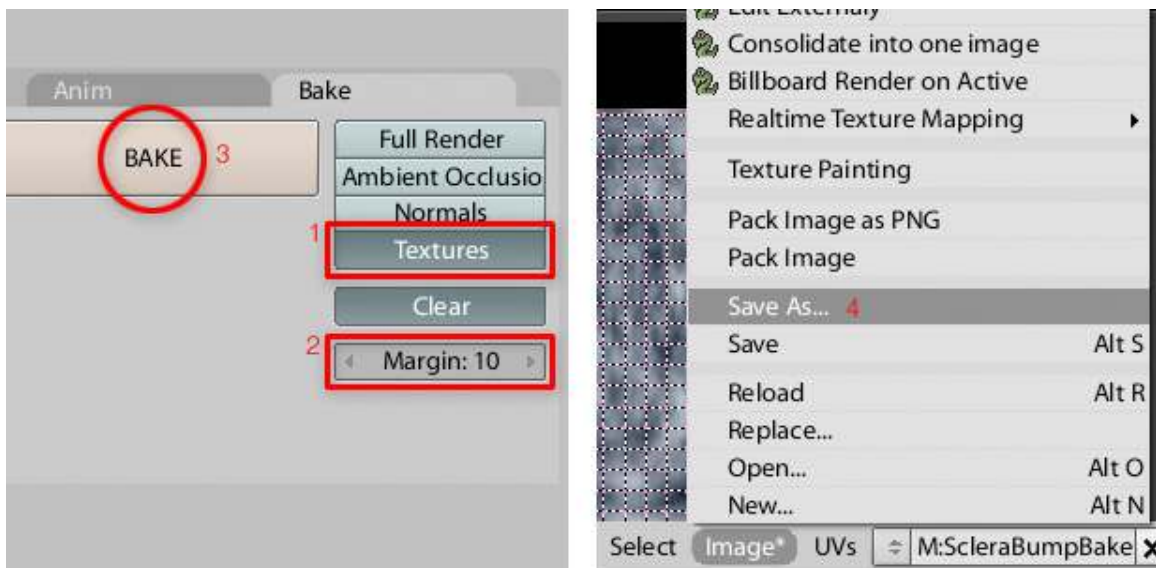


Figure 708

Now would also be a good time to **save your .blend file**.

(Where's the rest of this tutorial?)

Part III.

Unit 3 - Broadening Horizons

82. Beginning Lighting

Lighting, you say? Pssh. Just throw up one light source and let her run, right?

Wrong. Lighting is probably the most underestimated part of a scene by new 3D artists. By the following tutorials, you will gain knowledge of the technical use of lights in your scenes.

83. Understanding Real Lights

In order to learn how to light a scene properly it is helpful to first learn a few things about real lights. When you add a light to your scene you need to understand how to go about emulating the properties of the real light source that you want your Blender light to replicate.

83.1. The properties of real lights

83.1.1. Color

Every light source always has a distinctive color of its own. A photograph taken at sunrise or sunset is very easy to distinguish from one that is taken at noon just by looking at the color of the light. Incandescent light bulbs emit a light of a color that is different to that of fluorescent light bulb. So a scene lit with one of the two will have a different color cast when compared to the other.

83.1.2. Angle

Light always streams in from a particular direction. The angle or direction of light has a particular influence on which planes of an object receive light and the shape and direction of the shadows that are cast by objects receiving light from that particular source.

83.1.3. Brightness and decay

As light travels through space it decays or grows dimmer with distance. Think of a firework bursting in the night sky. When the firework first bursts the sparks are bunched close to each other. Over a short period of time they quickly spread out from each other. If you replace sparks with photons of light and the firework with your light source than you have a very rough analogy to how and why light decays. All real lights decay with distance according to the inverse-square law. At times when lighting a Blender scene it becomes necessary to cheat this law.

83.1.4. Shadows

Real lights will always cast a shadow of some sort. Shadows cast by all real lights are usually divided into two distinctive parts: a dark umbra, the inner core of a cast shadow and a softer less distinct penumbra. The ratio of penumbra to umbra determines how hard or soft a shadow looks. Hard shadows are usually cast by bright lights while on the other hand soft shadows are cast more by diffuse sources of light. All real light sources are soft to some degree. The hard clean and unrealistic shadows one gets from unadjusted ray traced shadows are the mark of poorly lit scene.

These are not all the properties of real lights, but as far as CGI lighting is concerned they are the most important. As we will see later on, Blender offers enough controls for you to tweak and adjust each of these key properties of light.

84. Adding Lamps

You can quickly add several different types of lights to your blender scene

SPACE > Add > Lamp > Spot

Noob Note In Blender versions 2.5+ you have to press Shift + A to get to the add menu.

A light will appear in the location of the 3D cursor. You can move a light just like any other object.

If you want to quickly light a scene just for illumination, not for a specific look, add four lamps around your subject. If you are interested in experimenting with a lighting arrangement, a nice quick way to experiment is to create a Monkey in the scene to test with.

SPACE > Add > Mesh > Monkey

The monkey is just as good of a test subject as a human face, so give it a try. You can throw various materials on the monkey and try different textures too. Don't bad mouth the monkey, she is really useful.

84.1. Explaining the Different Lights:

- **Lamp:** Single point light source. This simple light source can cast sharp shadows, but is best for filling in dark areas without shadows turned on.
- **Sun:** A light with parallel rays that will illuminate the scene with a bright even light. Often a good starting point for outdoor scenes.
- **Spot:** Spot lights produce light in a cone shape, and have some special features. They are the only light source that can be made visible with the 'halo' option, to simulate light in a fog. They are also the only light source that cast fake shadows, called buffer shadows. Unlike raytraced shadows, these buffer shadows are smooth, but not as accurate. They are FAST though, the only real choice for animation.
- **Hemi:** 180° constant light source. Great for use as a fill light, or as a back light.
- **Area:** These are similar to lamps, except that they are rectangular instead of an infinitesimal point. As a result they can cast accurate raytraced soft shadows. These are my lamp of choice for still shots, but they take a lot of time to render.



Figure 709 Light from Lamp



Figure 710 Light from Area



Figure 711 Light from Spot



Figure 712 Light from Sun



Figure 713 Light from Hemi

You might want to see the [adjusted scene](#)¹ and play around with it for awhile.

84.2. Creating a basic scene with basic lighting

- This addition is simply a way to apply what you know about lights and to discover a few settings like colors or creating simple shadows. The purpose here is to create a basic scene with a sphere over a plane, nicely lit. You should already know the basics of Blender (creating a mesh, moving and rotating it, rendering).

84.2.1. Creating the scene

- Okay, let's start! Open a new file. Add a **UVsphere** of 32 rings and 32 segments. Exit **EditMode**. Leave it in the center of the scene.



Figure 714 Smooth button

¹ <http://www.mediafire.com/file/mfk4k39w6fu38j5/light-test.blend>

Tip : Go in the Editing buttons

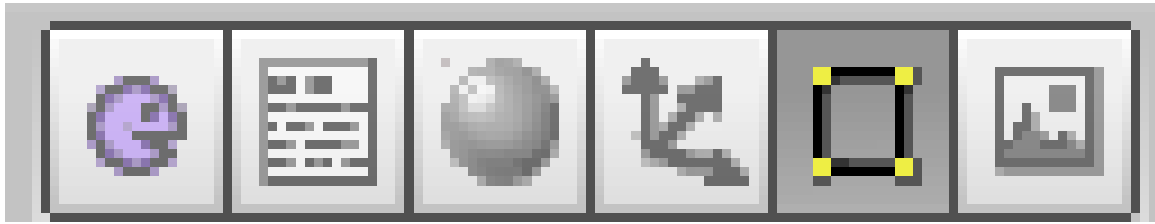


Figure 715 Edit button is pressed

, and push **Set Smooth** so the sphere will render as a nicely smoothed sphere.

- You should already have a plane in the basic scene, otherwise add a **Plane**. Have it in **ObjectMode** and move it just under the sphere. Scale it so it is very big. The ideal would that we can't see borders with the camera.

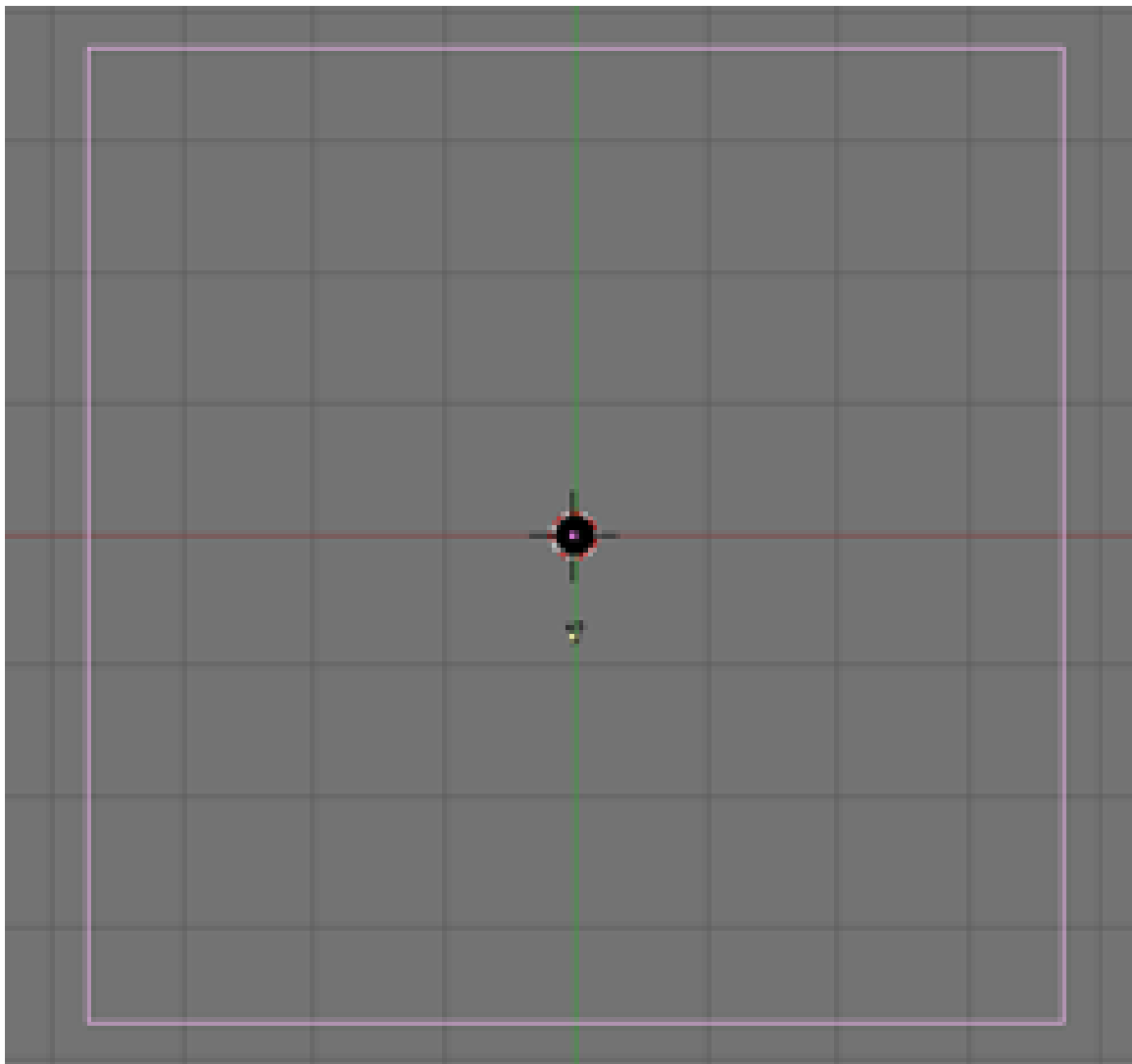


Figure 716 Top view of our REALLY BIG plane

- Then we will move the **Camera**. Grab it and rotate it so it looks at the sphere from top and a bit from the right. You can have an idea of what it sees pressing **Num0** to have a **CameraView**.

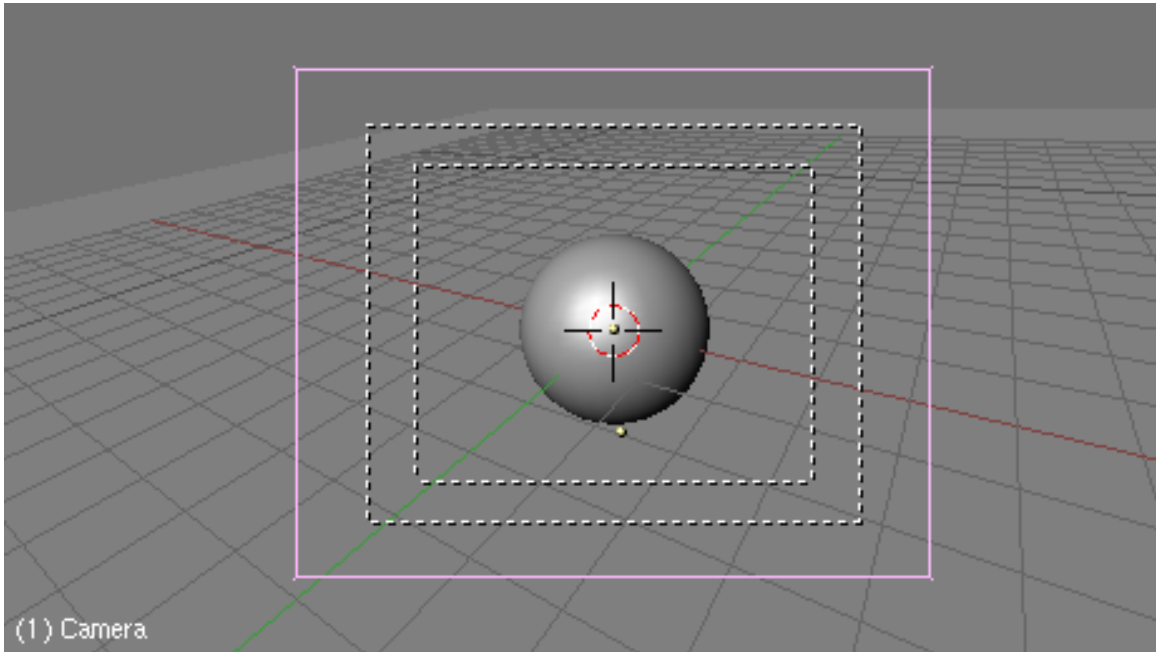


Figure 717 What the camera should be looking at

Trick : First click on the Camera, then on the Sphere holding **Shift**(the order is very important). Do a **Ctrl+T** and select **TrackTo Constraint**.

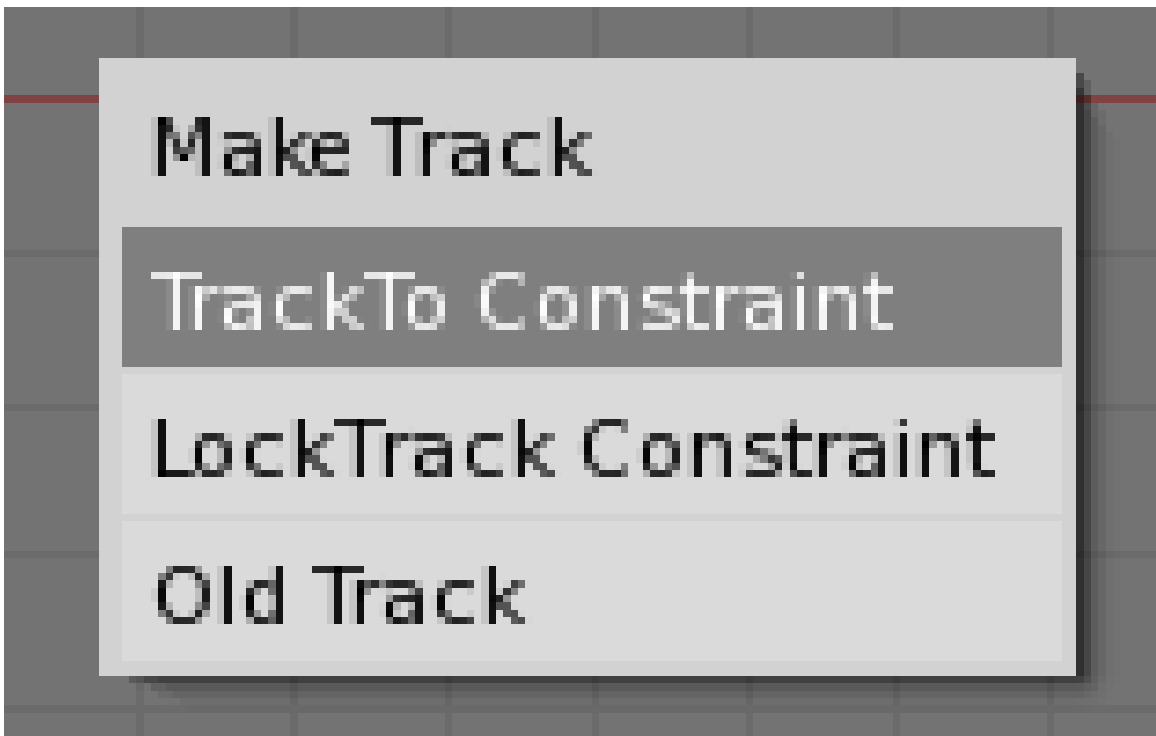


Figure 718 Select TrackTo Constraint

The camera will be looking at the center of the sphere... You can then move either the camera or the sphere and the camera will still point at the sphere.

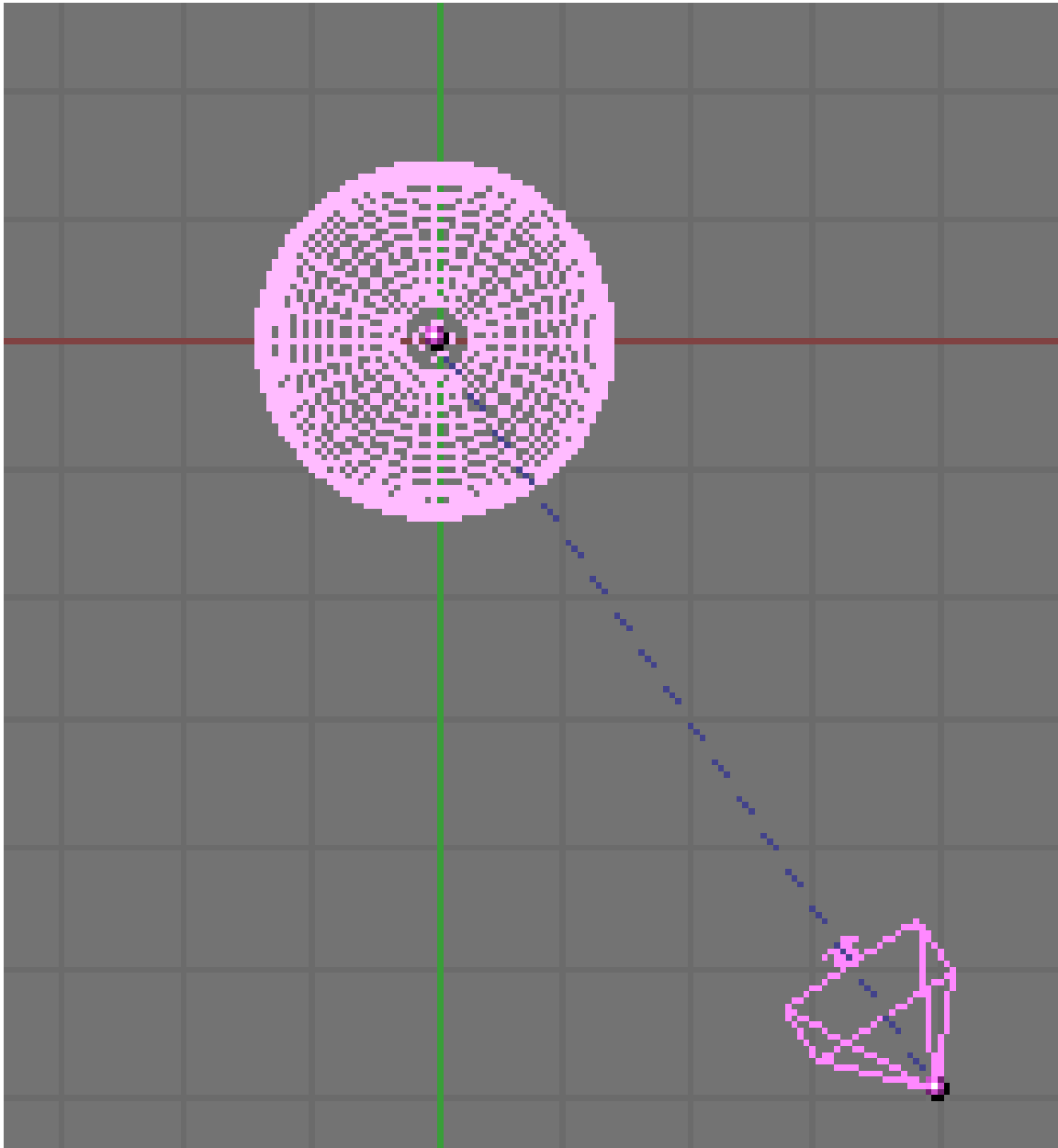


Figure 719 The Camera is looking at the Sphere

84.2.2. Adding the lights

- Okay, what we have is a pretty beautiful scene, isn't it?... Well, it isn't! But it is enough to add some lights... Here I will describe a basic lighting scene I use as a default for fast renderings. I picked it out from another tutorial, you can find the link at the end of this page.

- Now we will add our so-awaited lights. **Add>Lamp>Spot**. Yes, we will first use the Spot light. We can see it as a projector. This is the only light casting shadows. Place it so it is upper and on the left of the sphere. Rotate it so it looks at the Sphere (you can use a **Trick I** gave you before to have the light looking at the sphere).
- Okay, let's see what we can tune with the spot light. Having the lamp selected, go in the **Lamp** settings:



Figure 720 Click on Shading then on Lamp

You will see these buttons:

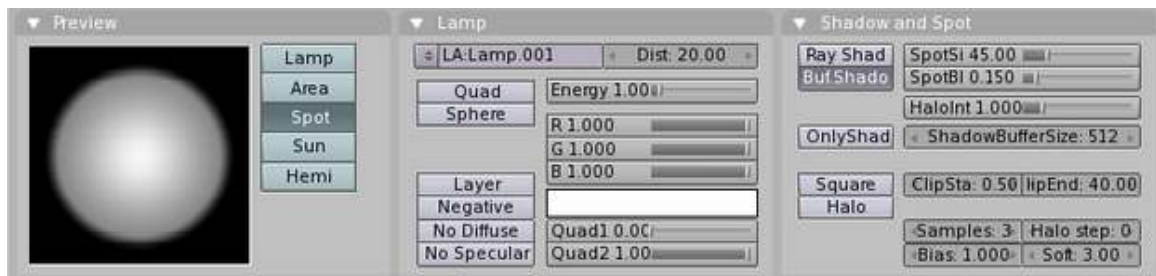


Figure 721 Lots of buttons, but don't worry !

- Yeah, really lots of options. Don't worry, I'll explain the basic ones.
 - **Dist** : Sets the maximum distance the light can reach. Increase it so the lighting cone really goes behind the sphere. I set it to **Dist:40**.
 - **Energy** : This is the force of the light. You can leave it at **Energy:1**.
 - **RGB** : You can change the colour of the light. Click on the colour and a little window will appear to select the colour you want. Leave it **white**.

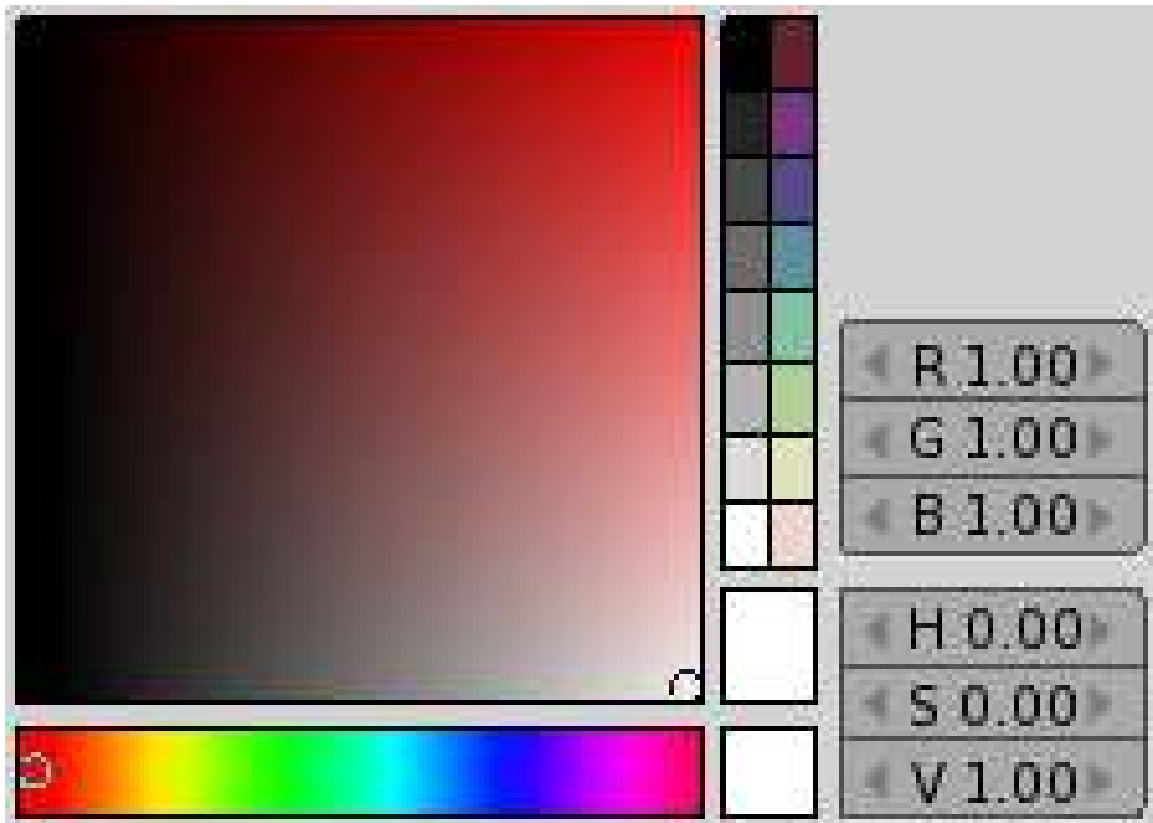


Figure 722 Let's colour our lives

- • **Buf.Shadow** : Enables the light to cast shadows, leave it **pushed**.



Figure 723 Here you will create the dark side

- **OnlyShadows** : This light only creates shadows, without casting light (yeah quite unrealistic, but it can be useful). Leave it **unpushed**.

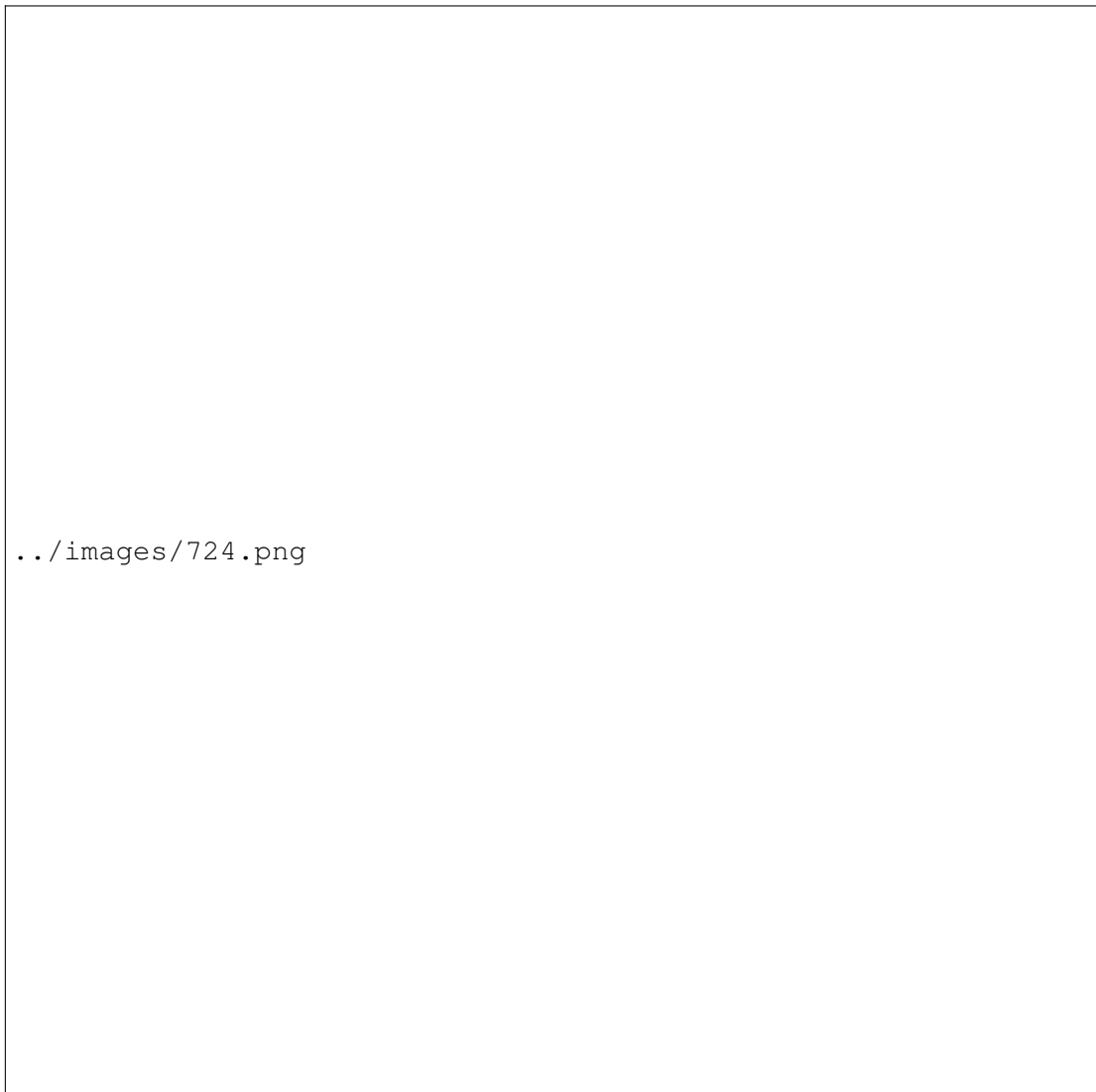


Figure 724 Adding a shadow without changing current lighting

- **SpotSi** : This is the angle of your cone, in degrees. Leave to **SpotSi:45**.
- **SpotBl** : This smoothes the circle cast by the light. We will smooth it so that it looks better. Set it to **SpotBl:0.4**.
- **ClipSta** and **ClipEnd** : This is the distance from the light between which shadows will be cast. You can see the "line of effects" in the 3D windows when you change these. Set them so the line starts before the sphere and ends far (well, a bit !) behind the plane. You should obtain something like this. (this is very important... If you do this badly, it can result in completely dark scene!)
- Now, we have set our Spot light. This light will be our side light and shadowing light. You can make a fast preview pressing **F12**. You can see your so nice shadow. But there isn't enough light... Let's add some more!

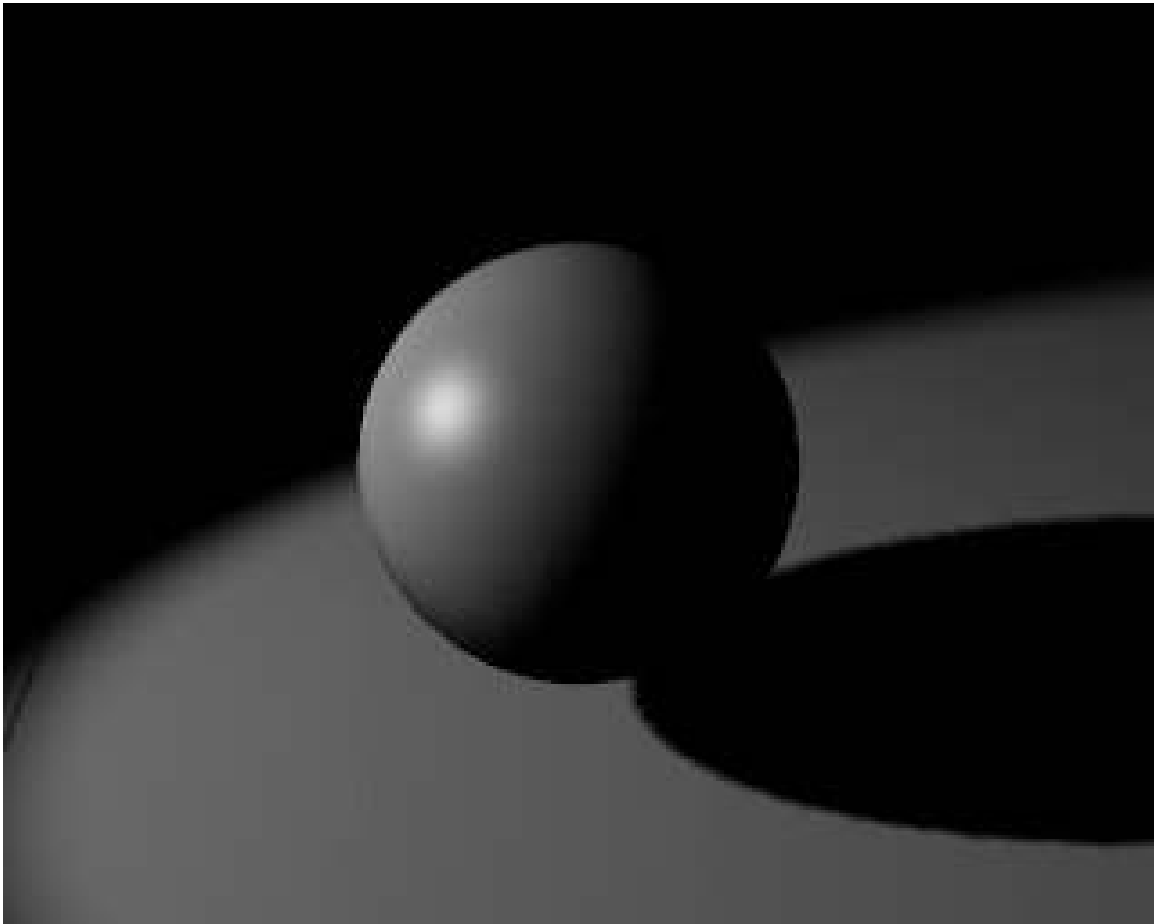


Figure 725 So dark! Let this tutorial bring light to your scene

- Time to add a second light! **Add>Lamp>Lamp**. This time, we will create a basic lamp. This is like a point which emits light in every direction from that point. You should place it at the opposite of the camera, quite at the same height. This light will be used to better see the form of the sphere and to add a sort of general lighting of the scene.
- Go in the Lamp buttons, and set it to a higher energy like **Energy:1.25**. You can make a quick render to see how much this light is important to a scene.

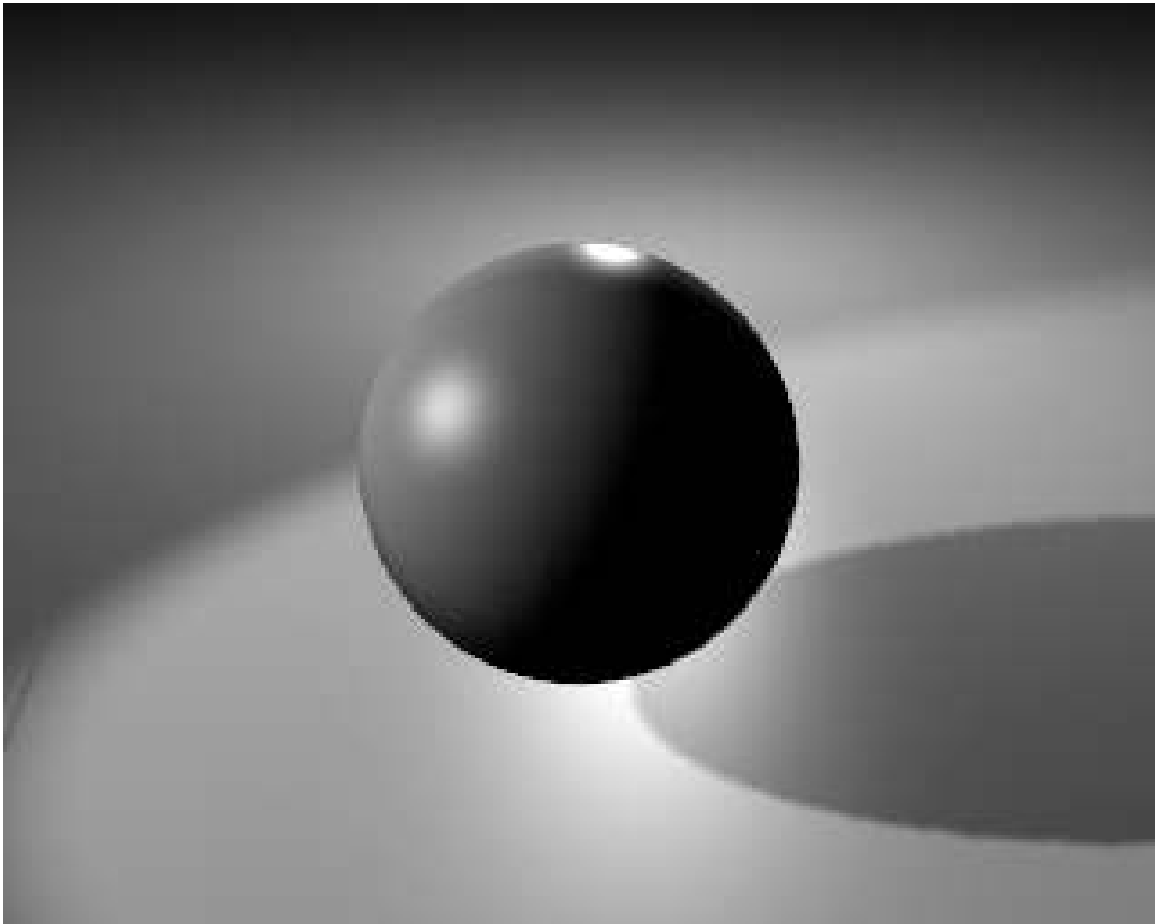


Figure 726 Yo, much better !

- Now, we will add a second basic lamp. **Add>Lamp>Lamp** and we will place it just behind the camera point of view, a bit moved at the opposite of the Spot light. This third light will slightly light the dark parts of the sphere.
- Decrease the energy of this light, as it is only supposed to fake the reflections of the environment. I set it to **Energy:0.8**. Another little trick, as this is not supposed to be a direct light, there shouldn't be a little white glow called **Specular** on the sphere coming from this light. Push the **No Specular** button.



Figure 727 This light won't cast speculars

- Okay, it's time for the final rendering. Of course, this is a really basic lighting set you can use for rendering a simple mesh; but for more complicated scenes, lights can come from other places, with other colors, etc... Thus we didn't use the **Sun**, **Hemi** and **Area** lights, which are a bit more complicated.

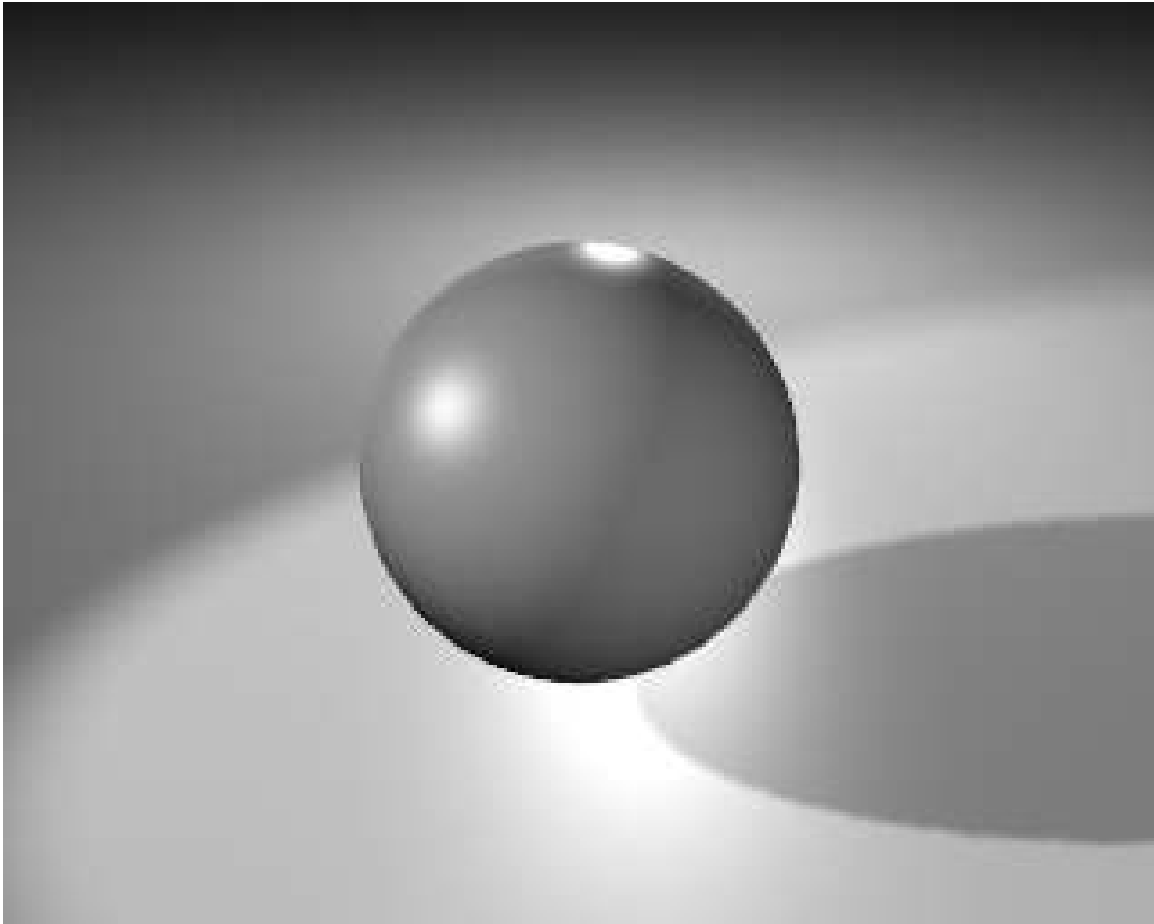


Figure 728 All this, only to light a sphere ...

Note that the following site contains nude figures: For a more in-depth tutorial, here is a tutorial from the Blender Documentation², which has been a great source of help for me.

User note: you might want to check this tutorial on lighting:
<http://www.youtube.com/watch?gl=US&v=YJiR2Q7uvbQ>

84.2.3. Outdoor lighting

- Here you will use a **Sun** in conjunction with a **Spot** light and some little **Lamps**.

² <http://download.blender.org/documentation/html/x4029.html>

84.2.4. Lighting Without Lamps

- It is possible to light a scene without lamps.
- Ambient Occlusion Video Tutorial: <http://www.youtube.com/watch?v=rKUAemD7oo4>
- Radiosity Lighting Video Tutorial: http://www.youtube.com/watch?v=_M4ffj2w_WY

85. Faked Gi with Blender internal

Getting uniform studio lighting with the blender internal renderer

85.1. Preamble

This tutorial will teach you how to use blender to create faked Global Illumination (here forth called "GI"). The reason it is faked is that blender currently doesn't support true Global Illumination but it's not a difficult task to fake. The results of this tutorial should look like this:



Figure 729

This tutorial assumes you know the basics of the blender interface and how to add objects, toggle editmode, and scale objects.

Now lets start blendering!

85.2. Blender Faked GI Tutorial

Add an ICO-sphere. Fire up blender and add an icosphere(**SPACE** → *add* → *mesh* → *ico-sphere*).Set the subdivisions to 3 and accept.

Scale the ICO-sphere by 15 times

Flip the normals of the icosphere so that they point inwards. Go into edit-mode and press the **A** key till all faces/verts/edges are selected and press **WKEY** and *Flip Normals*.

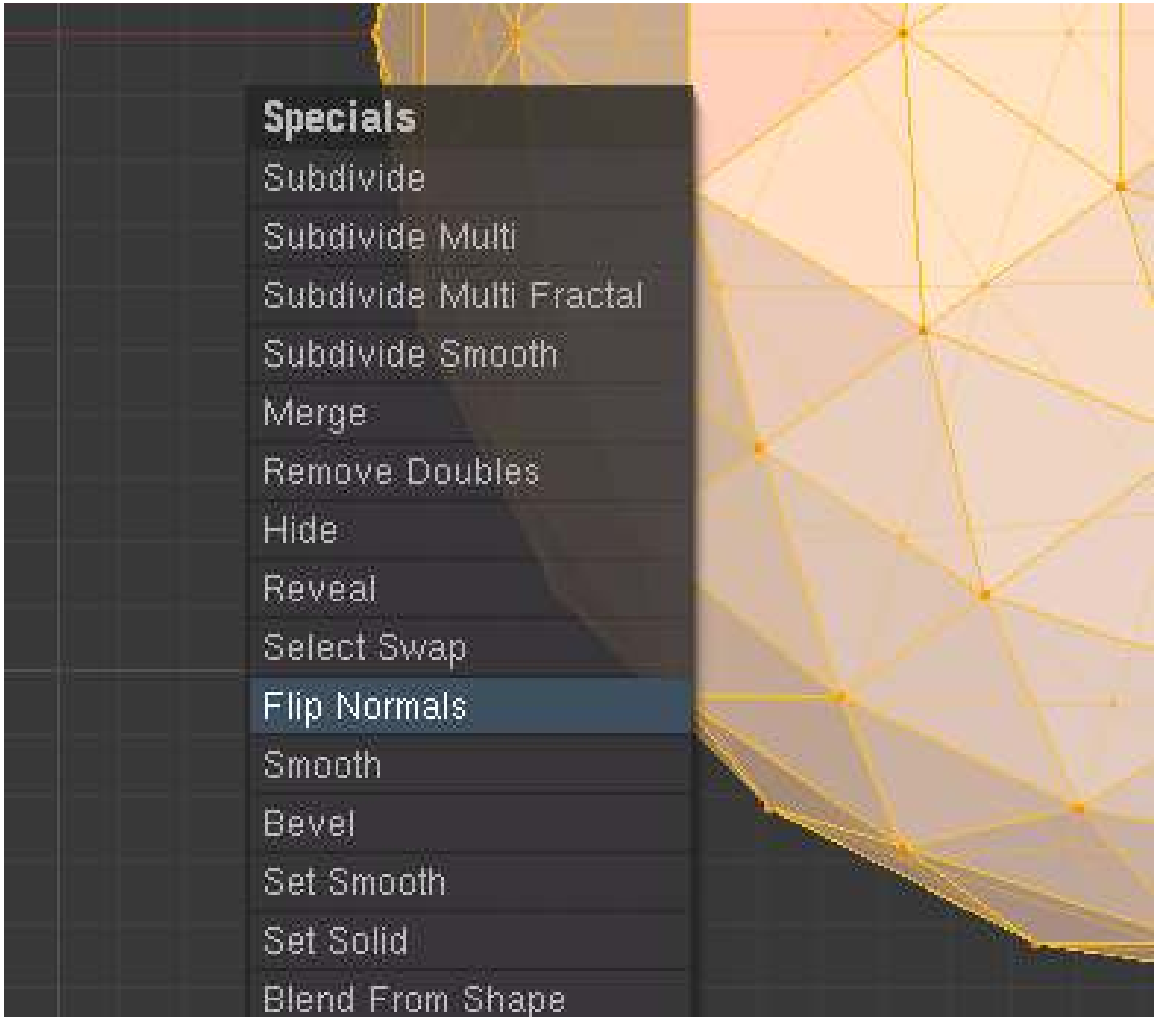


Figure 730

Add a sun light source (**SPACE** → *Lights* → *Sun*)

Set the energy value of the light. This requires some special attention. If you keep the value at the default 1, you get a pure white, washed out scene because we will duplicate the light 162 times. A good way to calculate the light intensity required is to mess with the one sunlight and do test renders while tweaking the energy value of the sun as required till you get the brightness you want. THEN you divide the energy value of your sun by the amount of vertices in your icosphere and then finally set the sun energy to the number you got.

Noob Note: I found I had to set the energy of the sun lamp extremely low, at .044. Otherwise, you end up with a completely washed-out image.

Parent the sun to the sphere. Select the Sun THEN shift select the sphere and hit **CTRL+P**
Enable dupliverts for the sphere. depress the DupliVerts and the ROT button.



Figure 731

Enable AAO (Approximate Ambient Occlusion) Set the AAO settings in the world buttons as in the following image:



Figure 732

Set up the scene to render This can be anything you want but I used a simple curved plane with Suzanne on top of it.

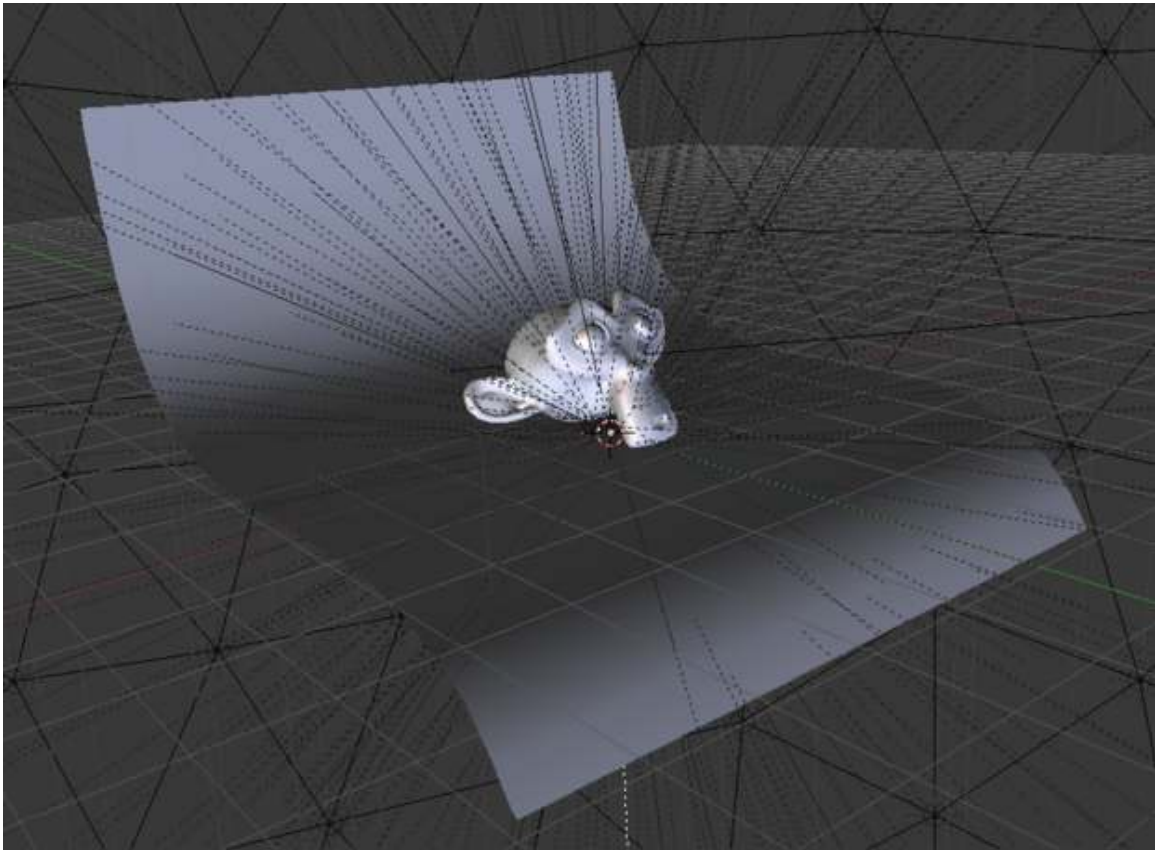


Figure 733

One more note: Often it is hard to see your scene what you have this huge sphere encompassing your scene so what you can do is to set the draw of the sphere to wireframe.



Figure 734

just one last note: if you want more directional lighting, select the sphere and go into editmode and press **CTRL+TAB+3KEY** for face select and select one face on a spot of the sphere and press **WKEY** and subdivided a few times to get more lights in one spot making that side a bit brighter and a stronger shadow on the far side of that cluster.

I hope you found this tutorial useful. If there are any errors in the tutorial, *Please* let me know, and I will fix them as soon as possible.

Blend happy and save often, --Radialronnie¹ (talk²) 22:47, 3 November 2008 (UTC)

-Discussions moved to the discussion page. Radialronnie³ (talk⁴) 15:15, 18 August 2010 (UTC)

-
- 1 <http://en.wikibooks.org/wiki/User%3ARadialronnie>
 - 2 <http://en.wikibooks.org/wiki/User%20talk%3ARadialronnie>
 - 3 <http://en.wikibooks.org/wiki/User%3ARadialronnie>
 - 4 <http://en.wikibooks.org/wiki/User%20talk%3ARadialronnie>

86. Render Settings

The render settings control various options related to the output of rendered, or full quality images. Rendering an image will calculate effects not displayed in the editing environment (due to their complexity), and therefore takes a larger amount of time to produce an image. If you notice an error partway through a render, then you can press escape to abort.

86.1. The Scene Tab

Let's take a look at all of the rendering options that Blender has. Open up a new scene (**CTRL + X**) and press **F10**. You should see something that looks like this:



Figure 735

These are the default settings for Blender. Let's take a look at all of these tabs and see what they do.

86.2. Output / Render Layers

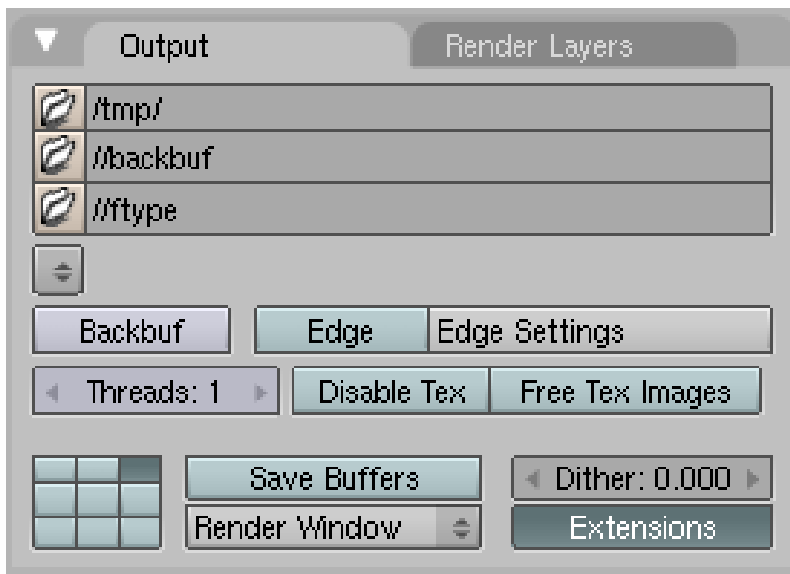


Figure 736 Output tab

This is the first tab on the left. First of all, we see a file locator icon (

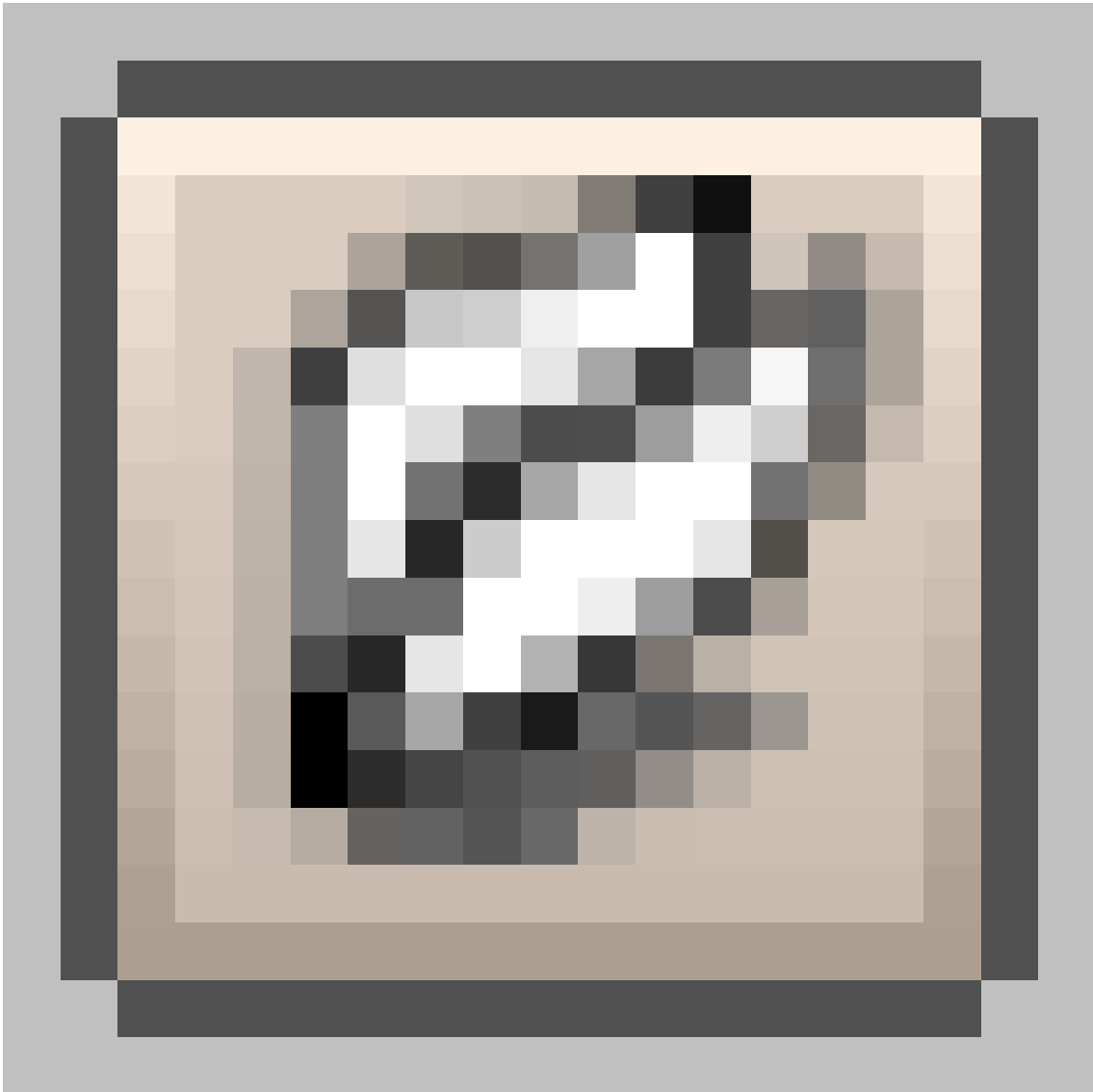


Figure 737

) to the left of /tmp/. This is Blender's temporary directory. (Normally C:/tmp on Windows) Below that, we see another file locator icon, labeled //backbuf. This is the Back Buffered image, one way to make an image the background on your render. Simply use that file locator to find a file, and Blender will use it as the background image (the backbuf button must be clicked to activate it). Finally, the last one is //ftype. The Blender description reads "image to use with FTYPE image type."

(Noob Note: There were some changes in v2.49: The Backbuff button has been moved to the very locator bar, to the right, as a blank button. Dither was moved to its place. And //ftype has been removed)

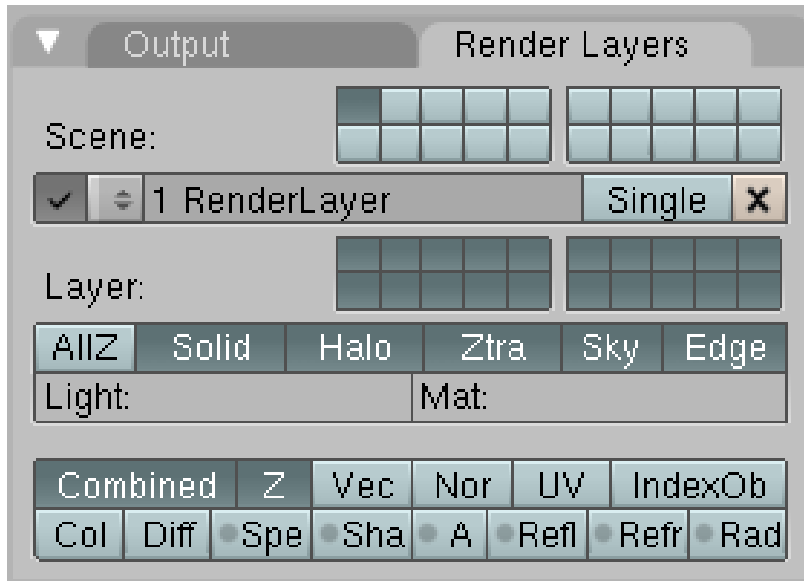


Figure 738 Render Layers tab

The next row down, we see this:



Figure 739

"What do these do?!" you ask. Well, it's actually quite simple. First of all, the first button is the backbuf button that I just talked about. If you have selected a background image to render using the "backbuf" option, clicking this button will make it show up. Next, on the right, are the 'Edge' and 'Edge Settings' buttons. Clicking the 'Edge' button will render your image with edges. Edges are rendered where the camera sees an edge. For an experiment, turn it on and render to see what happens. Notice how the default cube now has a black edge around it. If you want it bigger, click the 'Edge Settings' button. Here, you can change the size of the edge that is rendered, as well as the color. The next button we are going to talk about is the 'Threads' button. Clicking this enables threads, which allows the renderer to do threaded processes. Basically, if you have a dual-core processor (Athlon X2, Intel Core2Duo, etc.) it will have one processor render part of the image at the same time that the other processor is rendering another part. This (generally) helps reduce rendering time. Finally, the last button on this row is the 'Free Tex Images' button, which "Frees all images used by textures after each render".

There is one last row of buttons below.



Figure 740

The bunch of little squares enable you to tell Blender where to put the render window on your screen. Try pushing one and then rendering to see for yourself. The "Save Buffers" button saves all the tiles from every render layer and SceneNodes to files, saving memory. Experiment to try it out. The dither button will put dither noise in your image. The render window button allows you to render your image in a new window, render it full screen in the UV Image Editor, or render the result in the UV Image Editor (but not in full screen), a good idea when working with nodes. The extensions button adds extensions to your animations.

86.3. Render

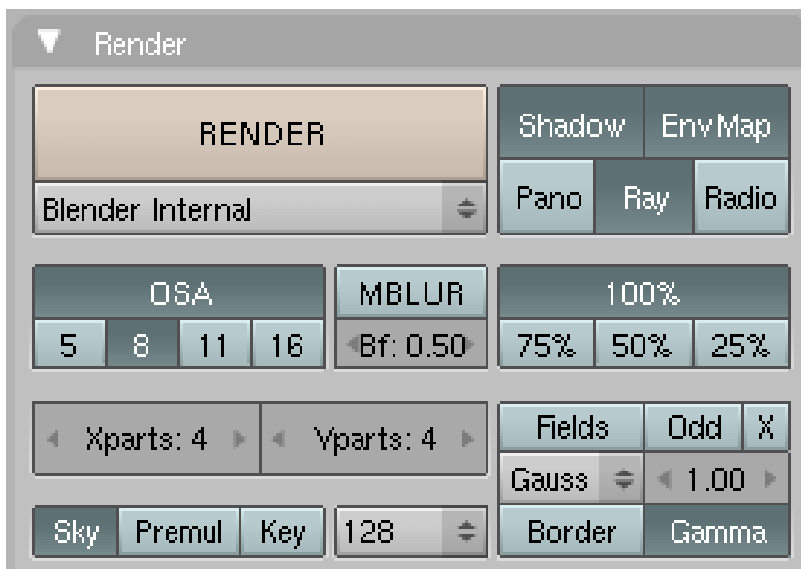


Figure 741 Render tab

Now, let's move on to the render tab! The first thing you should notice is the VERY BIG button that says 'Render' on it. This button performs the same action as the **F12** button. Below this is a drop-down box, allowing you to change between the Blender Internal Render engine and YAFRAY. (For more on YAFRAY, see YAFRAY Render Options¹). Next to the 'Render' button are 5 buttons: Shadow, Env Map, Pano, Ray, Radio. These are explained in detail below:

¹ Chapter 91 on page 799

- Shadow: Lights will cast shadows. If this is turned off, nothing will cast a shadow. To see this, add a plane below your default cube and render. Now turn this button off and see how there are no shadows.
- EnvMap: Use the environment map.
- Pano: This enables panoramic rendering. For more on panoramic rendering, see [Panorama Settings](#)²
- Ray: This enables Ray Tracing. Turn it off, and render the same images as I talked about in the description of the Shadow button. "What!?" you say. "That did the same thing as turning off the shadows! What's the deal?" Well, ray tracing is used to calculate how light goes through objects like glass and stuff. It also has something to do with shadows.
- Radio: This stands for Radiosity.

Moving on, the next button we come to is the 'OSA' button. For the use of this button, see [OSA](#)³

To the right of the OSA button, are some percentages. These will set the renderer to render at the specified percent of the resolution that you have set. (We'll come to resolution later) For an experiment, press the '50%' button. As you can see, Blender renders the image at 50% of the set resolution. This is an easy way to make a quick render.

Next on our tour, we come to two boxes labeled 'Xparts' and 'Yparts'. These buttons change the number of boxes that Blender will render in total. Play around with these to see how they change the number of small boxes in the render window when you render.

86.4. Anim / Bake



Figure 742 Anim tab

2 Chapter 89 on page 791

3 Chapter 88 on page 789



Figure 743 Bake tab

The final tab we'll (briefly) cover is the Anim(Animation) tab. If you have made an animation, pressing the 'ANIM' button will render your animation. Right below it are two buttons, Do Sequence and Do Composite. The Do Sequence button will render your animation with any effects you have done in the Sequence Editor and the Do Composite button will render an animation with your composite node work included. The Sta: and End: are the start and end frame numbers. The default setting is that your animation will start at frame 1, and will end at frame 250.

87. Output Format Options

87.1. Output Format Basics

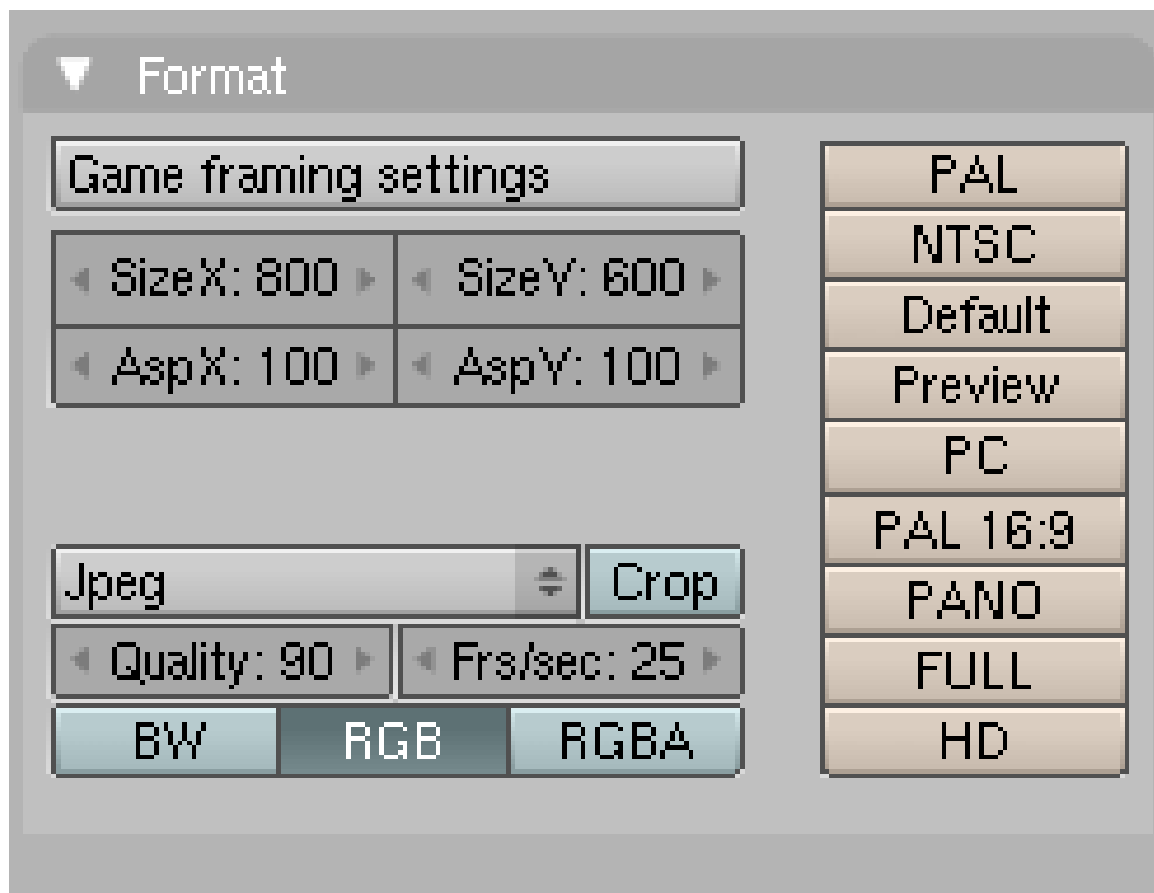


Figure 744

When you render a scene in Blender, by default a small window will open showing your image. If you want to publish your picture, you may want to render it a bit bigger. To do this, you have to open the Scene context (**F10**) and locate the Format tab (usually on the right).

You will see some settings there. Let's go through their meanings:

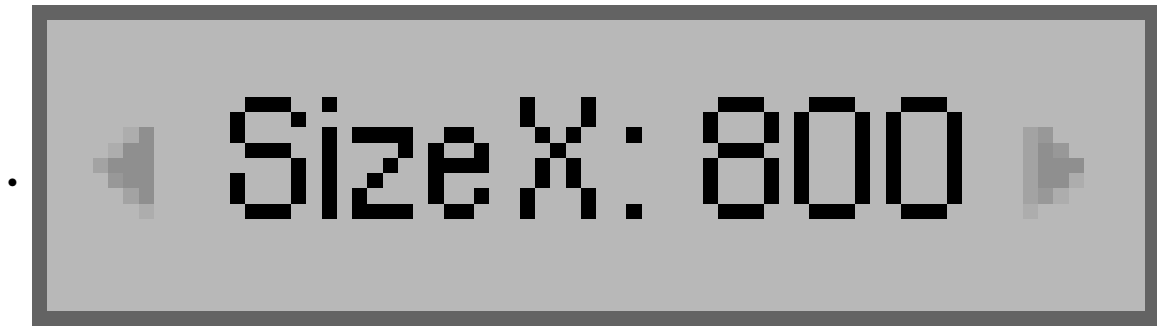


Figure 745

SizeX: This parameter sets the width of the image in pixels.

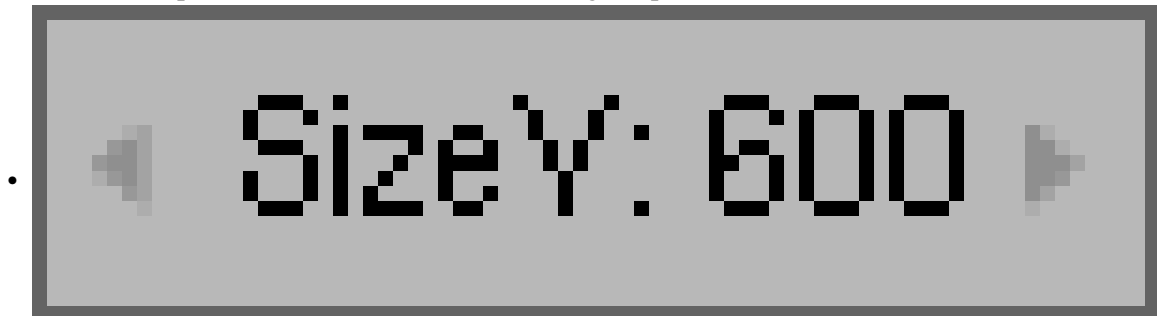


Figure 746

SizeY: This parameter sets the height of the image in pixels.

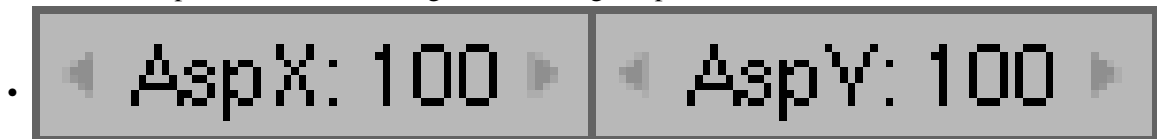


Figure 747

AspX, AspY: These parameters specify the aspect ratio of the pixels. By default, this is 100:100, because a pixel on a computer screen has equal width and height. These settings can be used for screens whose pixels don't have equal width and height. For example, on a PAL system one pixel's width/height ratio is 54:51, which you can select there easily. Notice that these parameters don't change the size of the image.

Don't forget, setting the number of pixels changes the resolution. Resolutions for your computer desktop are normally 1024x768, but can be larger. This means that it is 1,024 pixels wide and 768 pixels tall. To make an image this size in Blender, change the SizeX to 1024 and the SizeY to 768.

If this does not come out correctly, then you should right-click your desktop and click Properties, then go to Appearance. The highlighted numbers are X by Y.

87.2. Output formats

Blender has the ability to save in a wide variety of formats. To change the format, click the drop-down box (

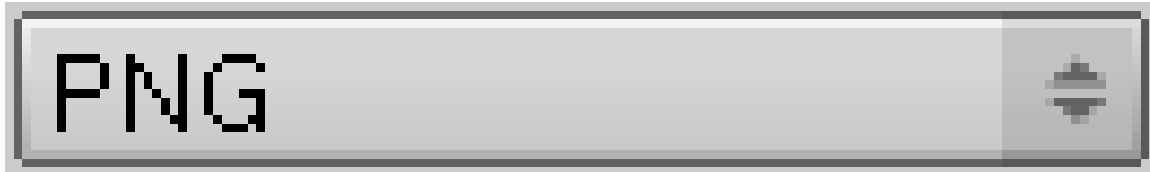


Figure 748

) halfway down the format tab. Currently, Blender can save files in the following formats:

- TIFF
- OpenEXR
- DPX
- Cineon
- Radiance HDR
- Iris:
- Iris + Zbuffer:
- HamX: extremely compact but only for the "Play" option
- Jpeg: default format. Lossy compression.
- BMP: uncompressed.
- PNG: open, lossless compression, alpha channel.
- Targa Raw: uncompressed Targa.
- Targa: Compressed Targa.

Single rendered images are not automatically saved to disk. Once Blender finishes rendering a scene (this can take some time), press F3 and the save dialogue will be opened where you can specify a filename and location for the output. You can retrieve the last picture rendered by pressing *F11*.

When working with moving images, you need to remember that you are often looking at file size (such as on the net) and compatibility with media players. Videos consists out of 2 elements: the file format and the codec. An example of this would be that you have a MOV file (the file format) with a H.264 codec. In other words, the codec is the way that the files are interpreted (compressed-decompressed = codec) by the media player. Youtube currently uses the FLV format with an H.264 codec. The VLC player should play almost any format available.

Video:

- AVI Codec: saves an AVI with a compression codec. Once selected a pop-up menu will appear giving options as to what codec you want.
- AVI Jpeg: saves an AVI as Jpeg images. Compressed but lossy.
- AVI Raw: saves an AVI with uncompressed frames.
- QuickTime: saves a QuickTime MOV file. Once selected, a pop-up menu will appear giving various video codecs to choose from. Can also remain uncompressed.

- FFMpeg: Implemented in a Google Summer of Code event, this codec is composed of multiple software libraries, so you can choose which codec to use under the video tab. The audio tab lets you choose from several different audio codecs to use and also has a bitrate setting. The Multiplex Audio button will render your animation with sound (if you have any), a nice feature.

For more information, see

- Blender 3D: Noob to Pro/Creating animated GIFs using Blender and Gimp¹
- Blender 3D: Noob to Pro/Creating Ogg-Theora movies using Blender²

87.3. Cropping

You can see



Figure 749

button next to drop-down box. If you activate it, image automatically gets cropped after rendering.

87.4. Palette



Figure 750

¹ Chapter 216 on page 1551

² Chapter 215 on page 1549

You can select three different palettes in Blender:

- **BW** - images are saved with BW (greyscale) data
- **RGB** - images are saved with RGB (color) data
- **RGBA** - images are saved with RGB and Alpha data (if supported)

87.5. Pre-sets Just For You

PAL	NTSC	Default	Preview	PC
PAL 16:9	PANO	FULL	HD	

Figure 751

The render settings in blender are customizable for unique results, but blender includes 9 pre-sets for common render outputs. One can use the pre-set as is or can use it as a starting point at which the settings are then modified.

PAL³ and **NTSC**⁴ are systems used in analogue Television. If someone were making an animation for TV or DVD, that individual may use one of these depending on which is used in their country. Both the PAL and NTSC pre-sets in Blender only affect the dimensions and pixel aspect ratio of the render and doesn't turn on anti-aliasing (OSA)⁵ or affect any other variable in the render. **Pal 16:9** is a newer an improved version of PAL which is also known as PALplus⁶.

The Default pre-set uses the dimensions and an pixel aspect ratio as PAL but also includes OSA⁷ and shadows.

The **Preview** pre-set generates a low quality setting for a fast render that usually is not used as a final output.

- **PANO** is short for Panorama. This setting allows for a full 360 degree panoramic render.
- **FULL** produces a render that fits your monitors full screen.
- **HD** is for High Definition.

3 <http://en.wikipedia.org/wiki/PAL>

4 <http://en.wikipedia.org/wiki/NTSC>

5 Chapter 88 on page 789

6 <http://en.wikipedia.org/wiki/PALplus>

7 Chapter 88 on page 789

88. OSA

OSA stands for *oversampling*, also known as *anti-aliasing*. If you look at the rendering below, you'll find that the one has a rugged edge. It looks like small steps! To stop this, Blender uses OSA to prevent "jagged edges" or "aliasing" as it's called.

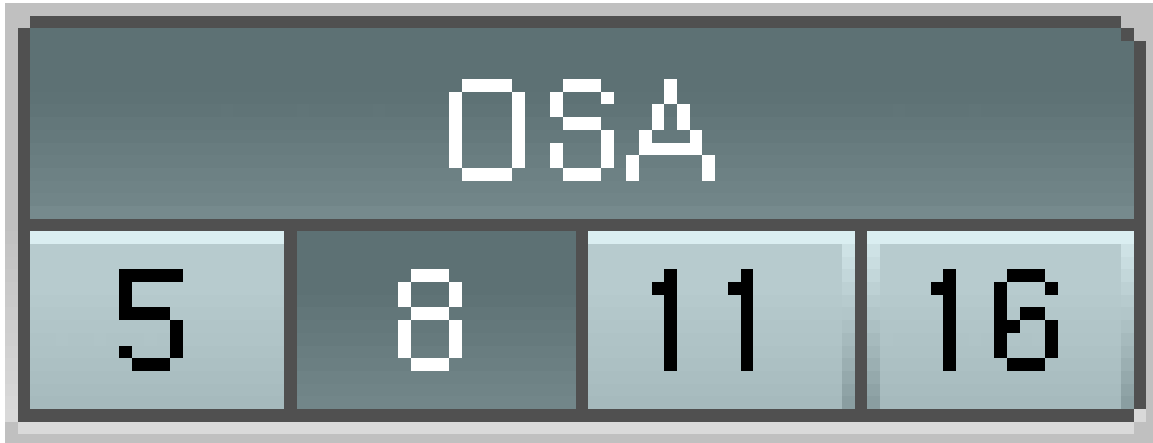


Figure 752 Oversampling settings in render tab.

This occurs when you have a diagonal change of colour, which results in rough edges. Remember drawing a diagonal line in Paint? To overcome this hindrance of square pixels, a technique called anti-aliasing or "oversampling" is used. It blends the colours around the rough edge to create a smooth, but defined edge. One way of doing this is to by creating the image twice as large, then scaling it down - oversampling. Blender can do this for you if you select an OSA rate. Remember this will take much longer, but results in better renders, so use this for the final product, not while testing. In some cases the scene can seem blurred due to oversampled textures; try changing the OSA setting, or oversampling yourself.

Here's a quick illustration of how OSA changes a render (look at the edges):

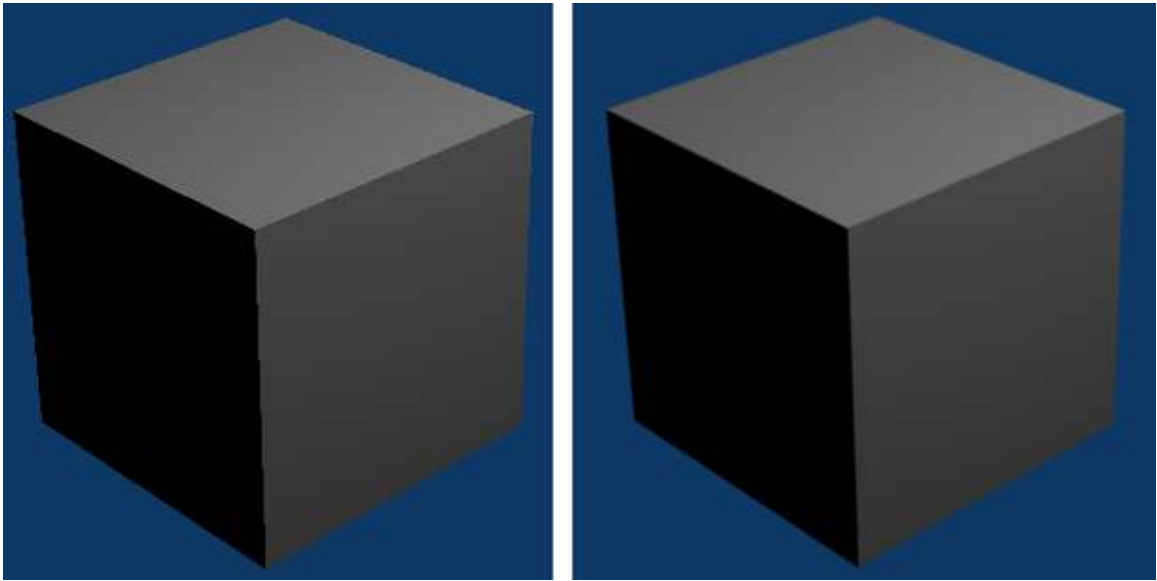


Figure 753 The image on the left has no OSA. The one on the right has 16x (the maximum amount allowed by Blender).

OSA settings can be manipulated in from the Render Settings panel.

89. Looking All Around - Panorama Settings

i Information

Applicable Blender version: 2.37a

89.1. Panoramic Renderings

89.1.1. This Article is Out Of Date

Panoramic features were changed somewhere between 2.37a and 2.42. More information, including how to actually use panorama, can be found in this thread in the forums: <http://blenderartists.org/forum/showthread.php?p=797886>

89.1.2. Building the Example Scene

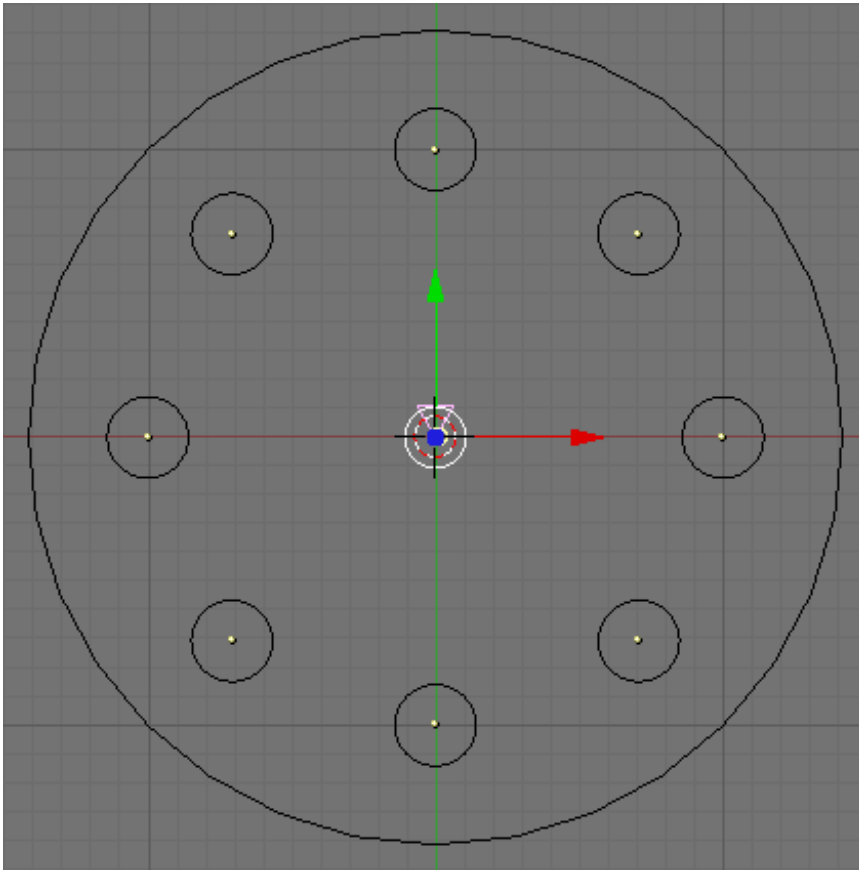


Figure 754 The example scene, see from above

Ever wondered how that cool looking 360 degree panoramas that you see on some websites are made? Well, I don't know either, so someone else will have to tell about that. What I found out though is how to make Blender output a 360 degree panoramic image - quite probably this at least is the first step when making something like on those websites.

Now, let's try to progress like in a tutorial, as in the other pages of this Wikibook. So, fire up Blender, and look at the well known initial box scene. Or rather, change it somehow. Assuming you have read the book in order, it shouldn't be hard to get something like in the picture.

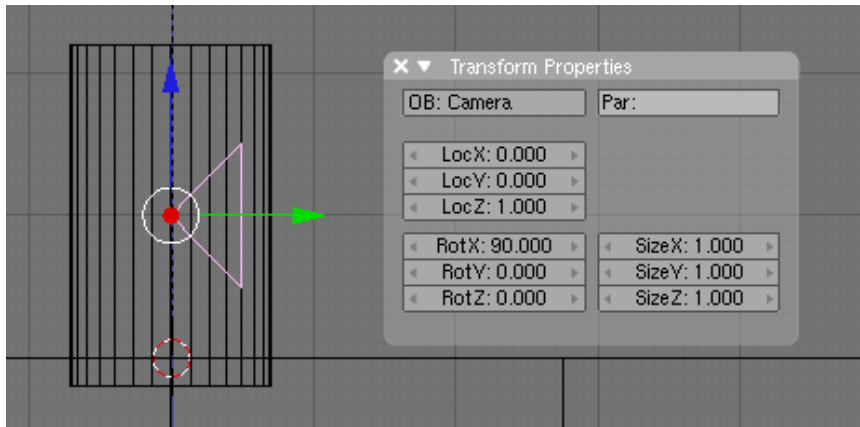


Figure 755 The example camera, seen from the side

I placed the camera and the light both above the origin, deleted the cube, and added 8 cylinders, all around the origin. One way to do this is to place one cylinder, then duplicate and rotate it around the origin. For this to work, set the rotation center to the 3D-cursor (with the Pivot button), and position the 3D cursor at the origin (use **SHIFT+S** to make the 3D cursor snap to the grid). Place the first cylinder using **GKEY** and then holding **CTRL** while moving the mouse. Change to object mode, then use **SHIFT+D** to duplicate it. Next, press **RKEY** to rotate it, then hold **CTRL** while rotating it by 45 degrees. Repeat the same for the remaining cylinders. Use **G** to move the light. Use **NKEY** to enter the camera values like shown, so it looks parallel to the ground.

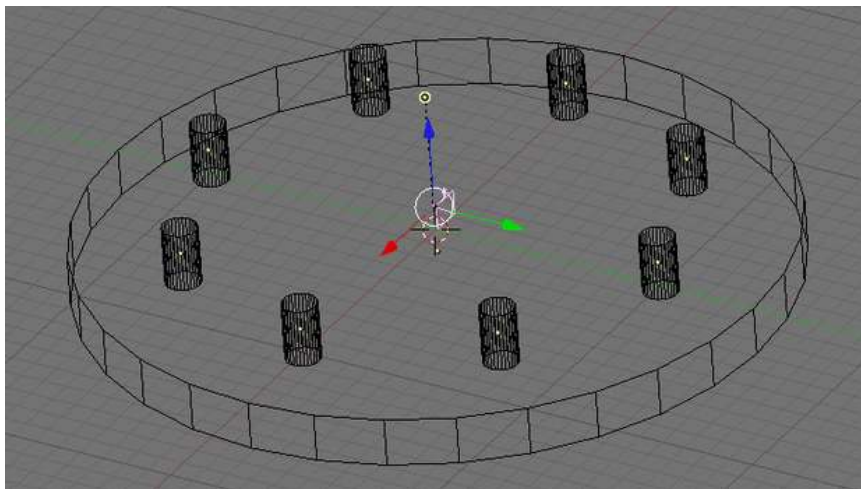


Figure 756 The complete scene with a ground cylinder

Speaking of ground, lets also add a ground. I added another cylinder, below the other ones, as shown. It's easy to add by switching to front view (**NUM1**), duplicating a cylinder (**SHIFT+DKEY**), and then moving it holding **CTRL** pressed again, so it snaps to the grid. To scale it, use the **SKEY**, but hit **SHIFT+ZKEY** to lock the z scaling, and hold **CTRL** pressed while scaling in x and y direction so you can snap to the exact size you want.

Now, to have a complete scene, we need some colors. I made the ground cylinder green, the 8 example cylinders gray, and also added a noise normal to the ground. This doesn't really matter here though, just needed some example scene for the 360° camera.

Something more interesting is the sky texture, because Blender's sky can be made to seamlessly wrap around with 360 degree. Belows is a screenshot of the sky settings I used. To get there, click on the Shading button, then select World.



Figure 757 The Shading->World Button

To change the texture, click the textures button or press **F6**, then add a new texture with the small button with two arrows, and select the type (e.g. "Marble" or "Clouds") instead of "None" for the texture. Go back to the World pane, and enable the Real and Blend buttons. And don't forget to use nice colors, I used blue and white. Now it looks a bit like white clouds in a blue sky.



Figure 758 The example sky

[Maybe should add here more details? What is important so you don't get a seam at the 0 - 360 degree point? How do you map pre-made sky textures?]

And we are already done, this is the example scene. All that is left to do is to render it as a panorama.

89.1.3. Panoramic Rendering

Go to the render settings (**F10**), and click the Pano button. Change the Xparts value to 4. The value in Xparts will tell how many times the camera will 'turn' horizontally when rendering. The Yparts value would do the same vertically. Each part (Xparts or Yparts) will render the size of the whole image you set. To make things easier, make the output image size quadratic and set both X and Y Aspect to 100. E.g. make your image size 600x600, but not 800x600, otherwise the following will not work.

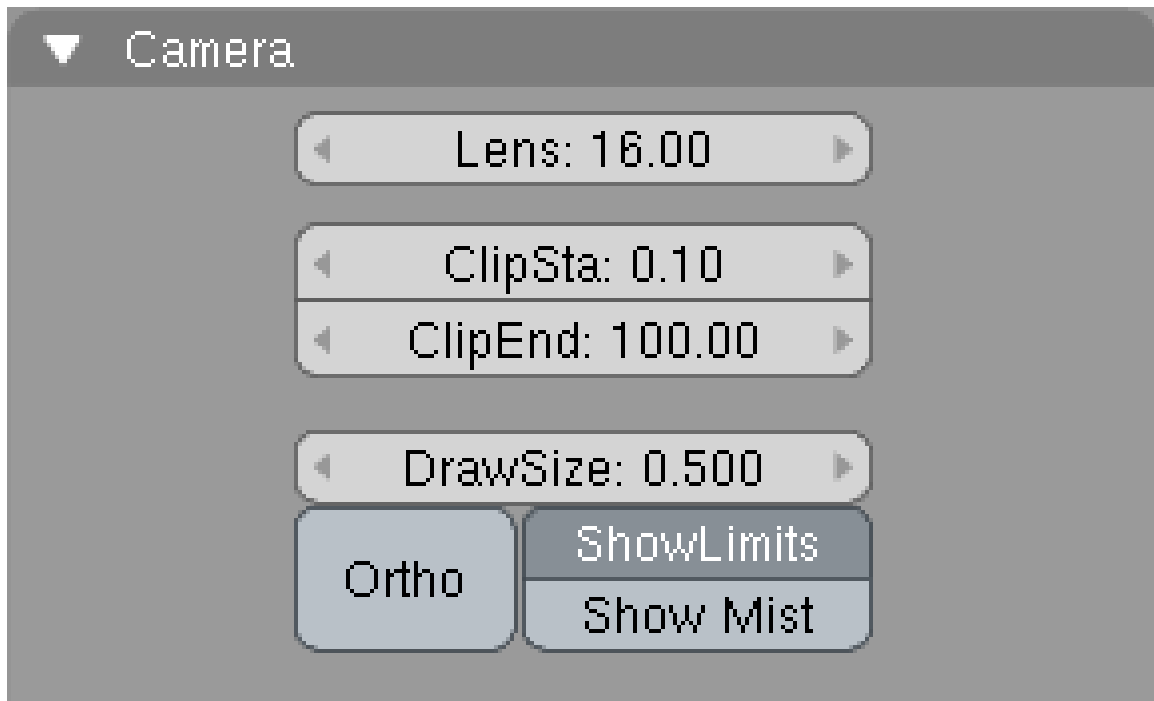


Figure 759 The camera settings

If you want a seamless 360 degree view, it is important to know how many degrees one single image spans. For example, if you know one image is 90 degree, then you can set **Xparts** to 4, and the result will be a single panoramic picture, and with the right panorama viewer, you can spin around in it endlessly.

But, how do you make a single picture 90 degree? The angle a picture spans is called field of view (FOV). And it is a property of the camera. To change it, select your camera, then click the edit button (or hit **F9**). Make sure you haven't set your camera to orthographic (the "Ortho" button), since FOV only exists with a normal camera.

You can directly enter the FOV in Blender's camera settings (as of version 2.42) - Toggle the 'D' button on, next to the parameter called **Lens**. Type in 90 to get a 90 degree FOV. Now hit **F12** to render our test image.

If you want, you can also try it out with any other scene. Place the camera somewhere in the middle with a good view all around, set the camera's **Lens** to 90, and go into the render settings (**F10**). Set **Xparts** to 4, enable **Pano**, and render. You should get an image 4 times as wide as a normal image, and the left and right edges should fit together seamlessly. Also note that Blender doesn't know if it should apply the FOV as horizontal or vertical viewing angle, but we always want it to be the horizontal angle. Therefore just make the single images of the panorama quadratic, as mentioned above - then horizontal and vertical FOV is the same and Blender gets it right.

This is the resulting panoramic image:

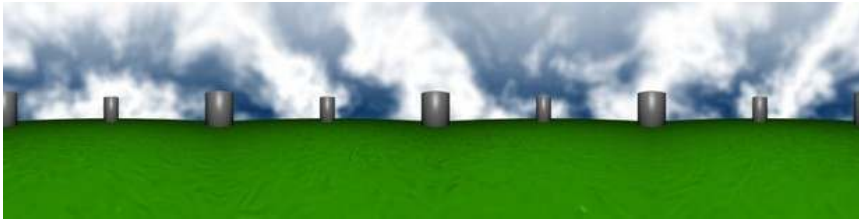


Figure 760 Panorama with 4 parts.

Warning: Didn't convert following lens settings to the new possibility with the 'D'-button for passing the FOV value directly. One problem remains, the image will be quite distorted. This is because a FOV of 90 degree is too high. Let's try with 45 degree and 8 XParts instead. As you will have noticed when playing around with the Lens parameter before, setting Lens to 8 instead of 16 will not achieve this, it will make the FOV even bigger. Setting it to 32 looks better, but definitely is not 45 degree. The picture to the right explains how FOV and Lens relate to each other:

The camera is at the bottom, the red angle is our desired FOV, and the length of the green line is what the Lens parameter represents in Blender. Therefore, the formula to calculate Lens when we know FOV:

$$\text{Lens} = 16 / \tan(\text{FOV} / 2)$$

For a FOV of 45 degree, we therefore get a Lens of 38.627. Now, write that into the Lens field in the camera settings, and render with 8 XParts. This time it will render an output image which is 8 times as wide as a normal output image, and again it will be a seamless 360 degree all around panorama of your scene. And this time with less distortion. With the above formula, you can now use any number of XParts you want - just divide 360 by the number of parts, and calculate the Lens parameter for the resulting FOV. Note that you also need to take the Aspect settings into account, unless you set it to 100 and create a quadratic output, as we did.

Some Lens settings:

- 4 parts: 16
- 8 parts: 38.627
- 16 parts: 80.437

The output with 8 parts can be seen below.



Figure 761 Panorama with 8 parts.

As always, feel free to modify this page in any way you want or add feedback to the Talk page.

90. Other Important Render Options

Rendering by Parts (Bucket Rendering)

Images can be rendered in pieces or layers rather than all at one time. Your computer will only need to compute smaller bits of information thus using less memory. By changing the Xpart and Ypart values (up to 8 each since Blender can't support more than 64 [8x8] parts) in the Render panel of the Scene context(F10,) you can divide your image into an invisible grid. The pieces will layer one at a time until they are whole.

Edge Renderings - Tutorial Here¹

Blender has an option of adding a border – a defined edge to objects (like in cartoons). To do that you need to go to the render buttons (F10) and then, under "output" change the edge setting: set the "Eint" to a value of about 100 and check the "edge" icon to enable edge rendering. This would give you edge line at the edge of each polygon. To prevent it from applying on all polygons the same way change to "unified render" under the format tab. Then, in the edge settings change the "antishift" value, it will decrease itself from the fint value when the line is between identical materials. By also checking the "all" icon you tell the render to apply the edge rendering on transparent materials as well. In Blender 2.59 this seems to be integrated into the Render-properties in the subwindow 'Post Processing' as options 'Edge' and 'Threshold'.

Note: User Terrywallwork - The edge bordering feature in Blender 2.43 only seems to work when Blender Internal renderer is used, when Yafaray is used, no edge is show even if the options is clicked. I'm not sure if there is a way round this or not.

HDRI Rendering - Tutorial Here²

Multi Thread Rendering

Blender has the option to render with 'Multiple threads' meaning it can render more than 1 part at once. This is done by sending more than 1 Render parts to the CPU to process at the same time, if your CPU can handle it. The Option is found in render buttons (F10). Change the number of threads used by changing its number.

Note in Blender 2.46+: Blender 2.46 automatically selects the number of threads.

1 <http://forum.nystic.com/viewtopic.php?f=34&t=3102>

2 <http://forum.nystic.com/viewtopic.php?f=9&t=5355>

91. YafRay Render Options

YafRay (Yet Another Free Raytracer) is an open-source alternative rendering engine supported by Blender.

Reader note: in the new version of Blender (2.46) internal renderer of Blender got a lot of upgrades and is now about two times faster than YafRay, so this article may be considered obsolete.

Yafaray 0.1.1 provides 5 different material types independent of Blender. It is also able to show objects behind transparent materials unlike Blender 2.49b.

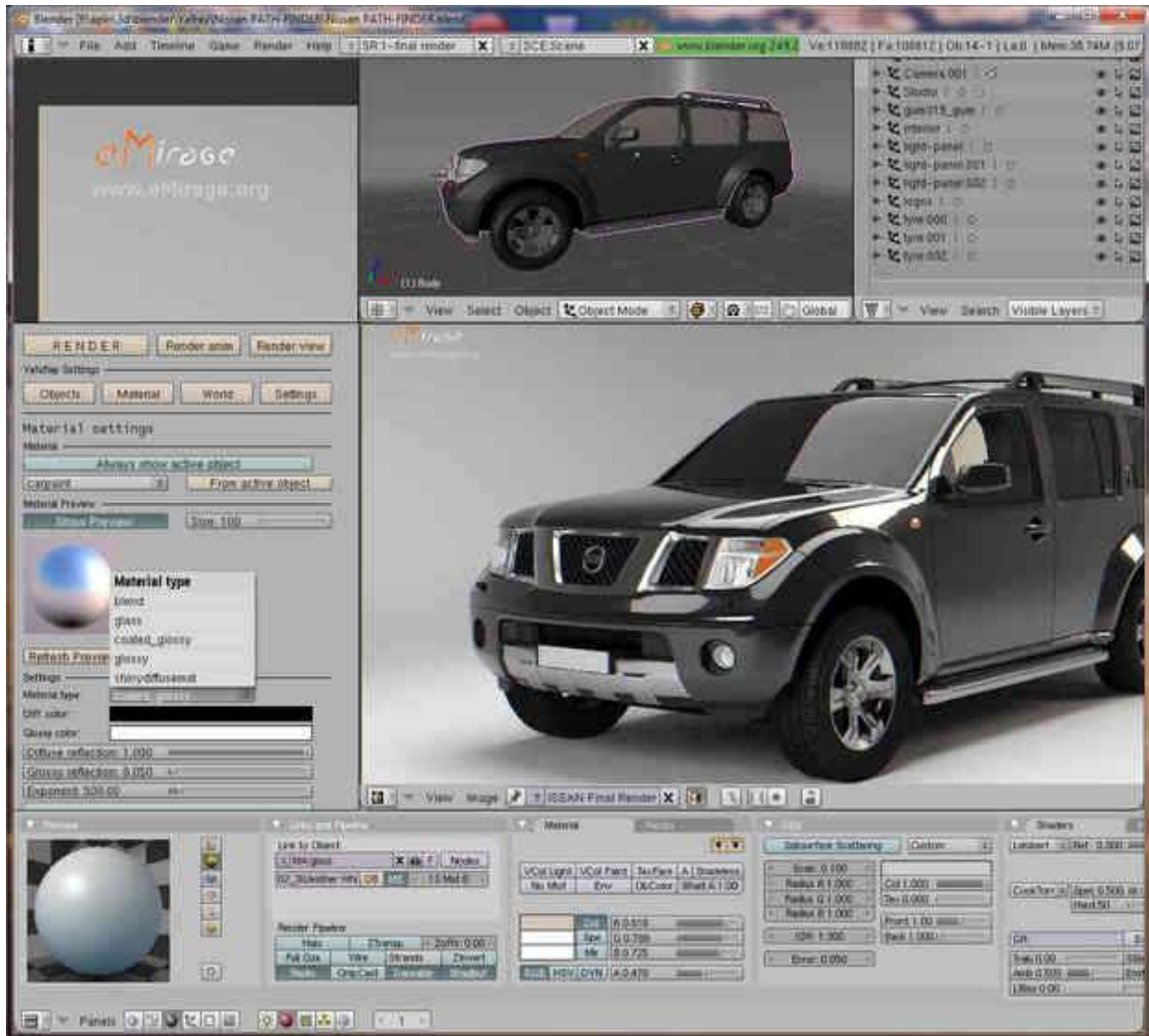


Figure 762

Further note: the Yafray project has been superseded by Yafaray¹. Yafray support was removed from Blender apparently around version 2.44. Use Yafaray instead. As of May, 2011, in MS Windows, it only works with 32-bit Blender 2.49b compiled with Python, with Python 2.6.6 32-bit version installed.

The YafRay render settings are normally hidden. To show them, switch from the Blender Internal rendering engine to the Yafray rendering engine.

- Go to the Scene Panel (Press F10)
- Click the Render Buttons button
- In the render tab, under the big Render button, click the Blender Internal button and select YafRay.

Now the YafRay and YafRay GI tabs appear.

91.1. Getting Started with Yafray

First, I would like to state that Yafray renders 2-3 times faster and nicer (includes Global Illumination, Caustics and...) than the normal internal built-in renderer in Blender. So it would just be worth it to download and install it.

On Windows, just download the .zip file and unzip it. You get the .exe file. Run it and follow default instructions. Latest version is 0.1.0 as of October 2008. Now change the Environment variable by adding C:\Program Files\YafRay to your PATH or whichever folder you installed yafaray.exe, users Windows 95/98/ME will need to restart where as WinXP users won't. Remember Yafray doesn't have a GUI and is a command line tool. Anyway you never need a GUI to run yafray as it runs nicely inside Blender as it is integrated into it. Go to the Command line and type yafray and press return. Now you have two paths

***Reader:** Can someone make this a little bit clearer? What is the Command line?*

***Answer:** The command line interface (CLI) is an interpreter for the operating system that provides a way of getting around by accepting commands from the keyboard instead of using the mouse to click on icons. It was the primary method of using the computer in MS-DOS, the predecessor to Windows. There are no buttons like in programs such as Blender. http://en.wikipedia.org/wiki/Command_line²*

You can access the command line in Windows by pressing Win-R then typing CMD and pressing the enter button. The command line can also be accessed by clicking on the Start menu, going down to "All Programs", over to the "Accessories" tab, and then clicking on the Command Prompt option. To change the environment variable, enter "path C:\Program Files\YafRay;%PATH%", without the quotes.

1) If there was an error regarding MSVCR71.dll, then download this file from the Web and place it in the c:\windows\system32 directory.

***Reader:** I have been using Blender for a while now, but due to my school's administrative limitations I am only able to use it on certain computers in school. For example, the computers that have already had 3d programs installed on them (like the ones in the Drafting and Design room) allow me to use*

¹ <http://yafaray.org>

² http://en.wikipedia.org/wiki/Command_line

Blender. Other computers give me an error message concerning MSVCR71.dll when I try to use it (specifically the ones in my Study Hall). Could you give me an alternate method of installing the .dll (maybe I can put it on a flash drive or in the My Documents folder)? **Answer:** Unfortunately, without administrative rights you will not be able to install a DLL. **Answer 2:** My school has the same type of administrative blocks, however I placed everything from my blender folder at home onto a flash drive, and now it works perfectly on all computers. Maybe try that?

2) Switch from the Blender Internal rendering engine to the YafRay rendering engine as explained and press F12 to Render. Wow, You should have noticed the lightning speed of YafRay.

91.2. YafRay tab

- **XML button:** If pressed blender will export an xml file that describes the scene and will pass it to YafRay. If not pressed blender will render with the YafRay plugin. When using the plugin interface, the render in progress will update continuously in Blender's render window. This gives much faster feedback. It even shows the distribution of the samples taken by the path light when rendering with cached pathlight. YafRay divides the image into blocks and renders them in a random sequence.

92. Stamping

92.1. Tictoon Tutorial: Stamping

Stamping in blender is a rendering option. A stamp on a render can prevent that render from being lost, keeping your project organized, and keeping your files organized.

An example of a stamped Image:



Figure 763 A stamped render

Now my render is associated with a file, I know the date it was rendered at, and I've stated that the image is property of the OWF project.

Let's start off by opening up Blender. We are presented with the default scene.

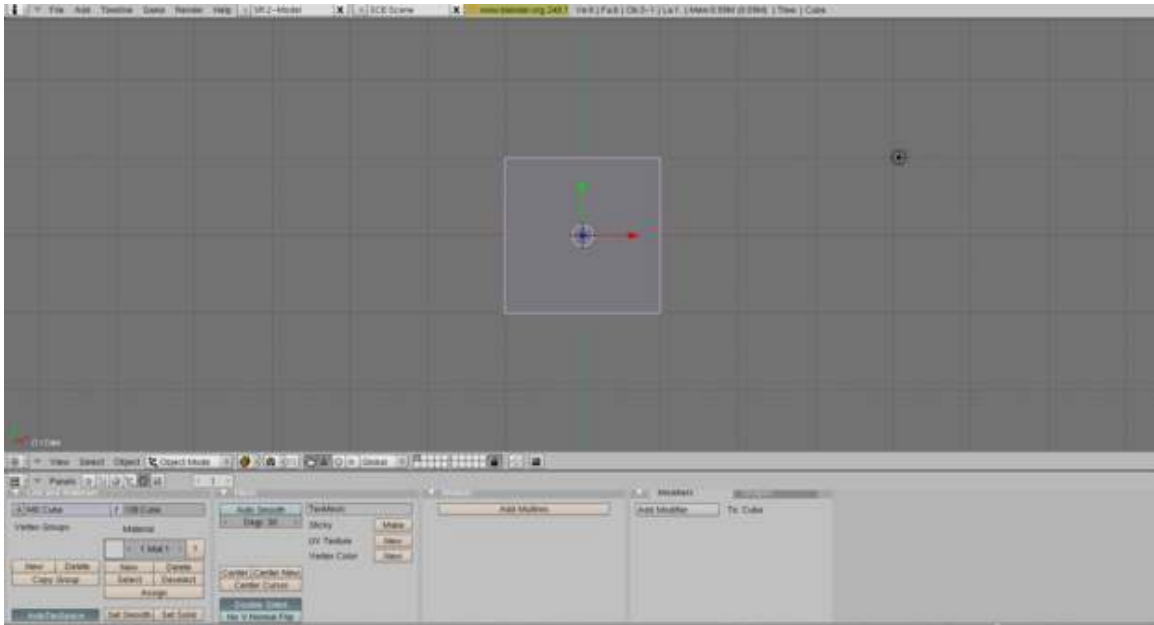


Figure 764 The default Blender Scene

Go to "Scene". Hit F10 or click the Scene button.

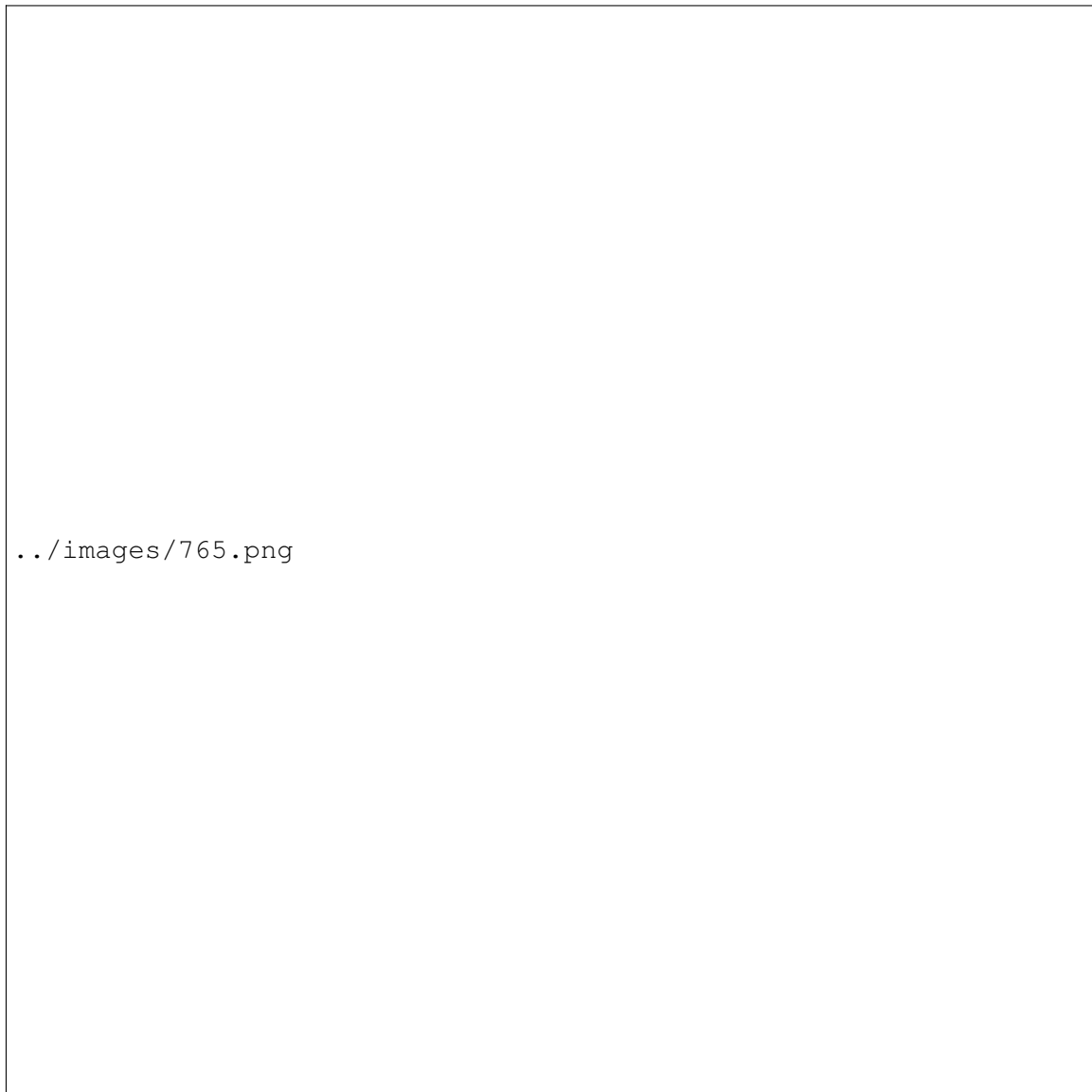


Figure 765 Scene Button

On the far right side you can spot a panel for the format of your renders.



../images/766.png

Figure 766 Format Panel

Pick the "Stamp" tab.

You should be presented with a button:



Figure 767 Enable Stamp button

Click on the button and you should see a variety of options.

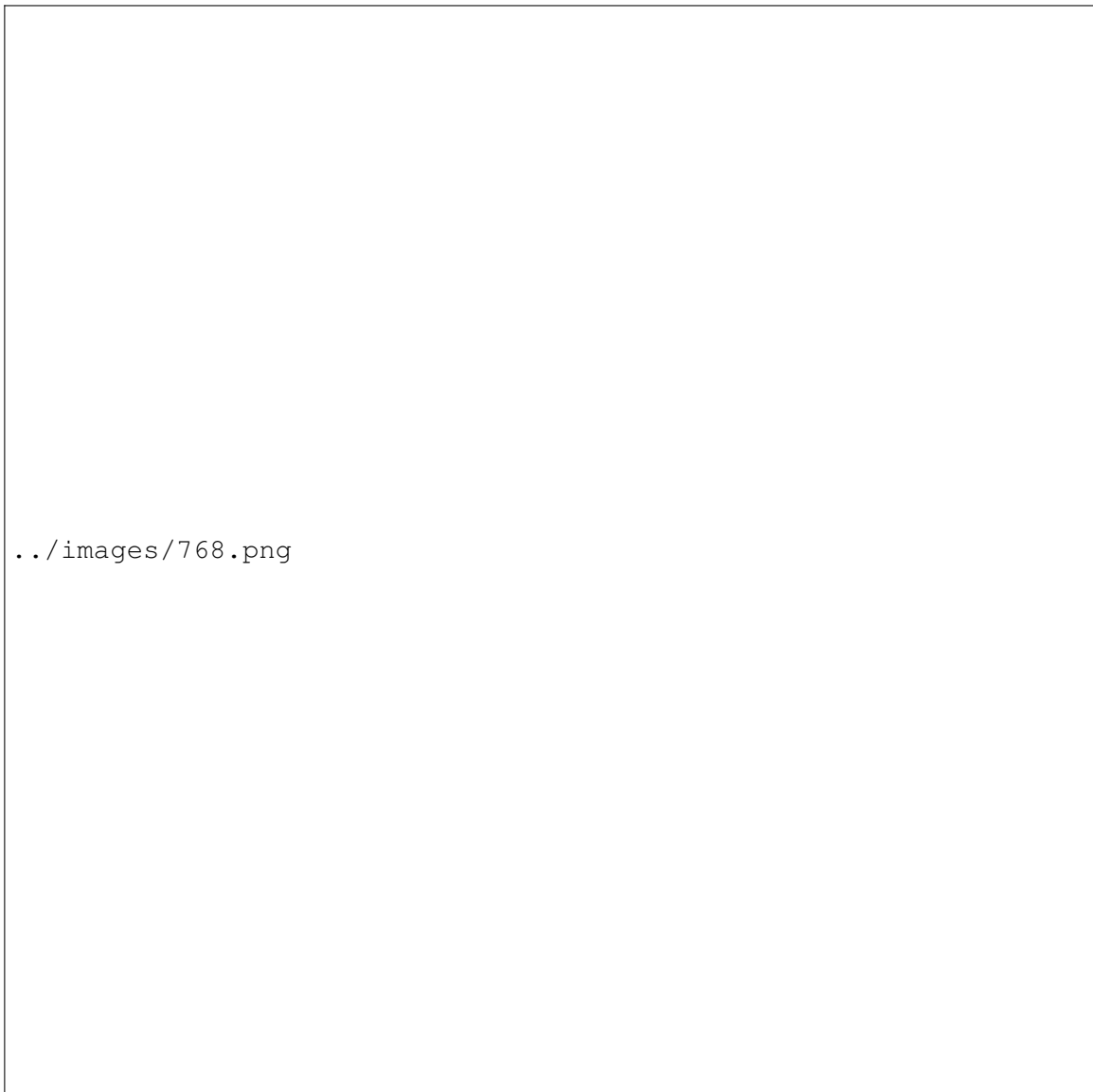
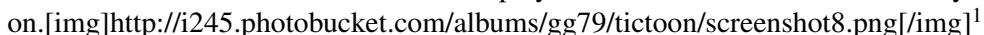


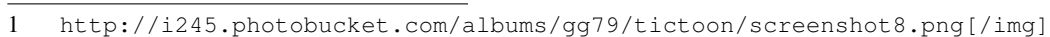
Figure 768 Stamping panel

Enable Draw Stamp to see your stamp in your render, more on Draw Stamp later on.

Each setting does something different. Here's a breakdown:

Filename: If enabled, this displays the path to your file. If you haven't saved your file yet, it will display "*File <untitled>*". Filename is displayed in the top left hand corner. When more than one stamps are in the top left corner of your render.

Scene: If enabled, this will display the name of the Scene your render is on. It will be displayed in the bottom right hand corner of your render.

¹ <http://i245.photobucket.com/albums/gg79/tictoon/screenshot8.png>

Camera: If enabled, it displays the name of the camera from which the scene is rendered. It is displayed on the bottom of your render in the middle.

Time: If enabled, this displays the time stamp of your animation. It is displayed in the bottom left hand corner of your render.

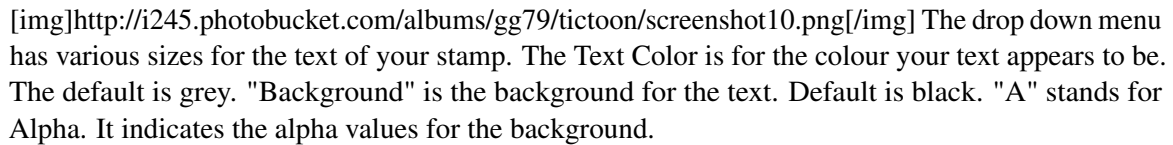
Date: If enabled, the date of the render is displayed. It appears in the top left corner of your render.

Frame: If enabled, the frame of your animation will be displayed in the bottom left hand corner.

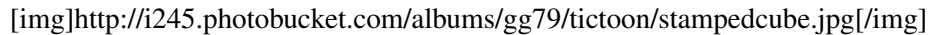
Marker: If enabled, the marker will be stamped onto the render. It is in the bottom left hand corner.

Sequence Strip: If enabled, it will display the sequence strip. Displayed in the top right corner.

Note: This is a custom stamp. you can type anything you want in here and it will be stamped in the top left hand corner of your render.

Draw Stamp: This needs to be enabled for the stamps to be seen in your render.  The drop down menu has various sizes for the text of your stamp. The Text Color is for the colour your text appears to be. The default is grey. "Background" is the background for the text. Default is black. "A" stands for Alpha. It indicates the alpha values for the background.

Now enable some buttons and hit F12 and you should see stamps on your render!

Efficiency tip: using this with Evernote can be very useful because Evernote can find text within images so you can quickly find a render if it has a stamp. I use the filename and date taken to find my images in Evernote. Keep blending :D 

93. 2D, Toon, & CAD Rendering

93.1. 2D, Toon, & CAD Rendering

Rendering a 2D Black & White Line Drawing. (ver.2.49, defaults)

1) *Background*. Under the menu selections, "Shading" (F5) → "World buttons" → "World" tab. Turn the background (default is blue) color to white. The color selection box above the "HoR" slider.

2) *Edges*. Under the menu selections, "Scene" (F10) → "Render buttons" → "Output" tab. Click the "Edge" button to turn on the edge lines. Adjust the "Edge Settings" if wish to change the edge intensity (sensitivity) & edge color. Under the "Format" tab of the same menu you can also choose to change the file format (default is Jpeg) & render size.

3) *Object Color*. Under the menu selections, "Shading" (F5) → "Material buttons" → "Links and Pipeline" tab. Select the materials to be rendered. Once a material is selected, move to the "Material" tab and change the color (default is gray) to white as well. It is the top box of 3, next to the 'Col' button. Each material instance will need to be changed.

4) *Shading*. Eh, who needs it? Menu selections, "Shading" (F5) → "Material buttons" → "Material" tab. Click on the "Shadeles" [sic] button to remove shading. Each material instance will need to be changed here as well.

5) *Camera*. Select the Camera in "Object Mode". Under the menu selection, "Editing" (F9) → "Camera" tab. Click on the "Orthographic" button, to flatten the image.

6) That's it folks. Hit F12 to render, and F3 to save your render.

**Note: Camera in Orthographic Mode:* Leaving the camera in Perspective mode will result in mesh lines being barely visible, almost white washed. Where as Orthographic mode will allow the mesh lines to become much more visible. And combining Orthographic mode with SubSurf, can help to create well defined curved surfaces. Just remember to change the camera lens scale, and Crease edges (Shift+E) where needed.

Edge Line Thickness

Overall edge line thickness can be changed by adjusting the render resolution. As such, the greater the render resolution, the thinner the edge lines will appear. Under the menu selections, "Scene" (F10) → "Render buttons" → "Format" tab. Change the "SizeX" & "SizeY" entries (Max = 10,000x10,000). For complicated images at high resolution, you may wish to change the number of tiles rendered. Found under the "Render" tab, "Xparts:" & "Yparts" (Default = 8x8).

Edge Line Feathering

Edge line weight can be adjusted by changing the planar distance, in the 3rd dimension, of objects relative to each other. Or in other words, how close an edge is to a common plane or an angle to 180 degrees. Resulting in edge lines appearing lighter (gray) or even totally disappearing.

Some gradient can also be created in lines by sloping an edge towards a common plane. Causing lines to fade from black to white depending upon the relative angle.

What's Needed: A way of defining edges with line type, line weight, & color using blender. Any suggestions?

Putting CAD in the Blender

Python Scripts Catalog: CAD Tools¹

(Article)²

How to add 3d dimension lines to Blender? (Article)³

2D External Tutorials

Rendering architectural line drawings with Blender (Great site for Architectural ref.)⁴

Toon Shading - A Blender Tutorial (video)⁵

Blender Line Drawings⁶

Blender Compositing Tutorial for 2D Animation (video)⁷

Tutorial 1 :Sort of sketch style rendering with nodes⁸

Tutorial 2: Pencil sketch rendering with nodes⁹

2D Scripts & 3rd Party Programs

VRM: Vector Rendering Method¹⁰

PantoGraph¹¹

Freestyle¹²

93.2. Freestyle & Blender Integration

Getting Started

- 1) Download & run a Freestyle build from **GraphicAll.org**
- 2) Under the 'Render' menu, open the 'Post Processing' sub-menu. Select the 'Freestyle' box.

1 http://wiki.blender.org/index.php/Extensions:Py/Scripts/Catalog#CAD_Tools
2 <http://www.blender3darchitect.com/2009/07/blender-3d-as-a-cad-tool-using-the-procad-script>
3 <http://www.blender3darchitect.com/2008/11/how-to-add-3d-dimension-lines-to-blender/>
4 http://www.opendimension.org/blender_en/line_drawings.php
5 <http://www.youtube.com/watch?v=nhaVsNqbs8M>
6 http://www.royalhigh.edin.sch.uk/departments/departments/CDT/ahgc_0607_blender_line_drawings.html
7 <http://www.youtube.com/watch?v=83nNEFMWr2Y>
8 <http://matthieu3d.free.fr/TutoViral/tuten.html>
9 <http://matthieu3d.free.fr/TutoVira2/tut2en.html>
10 <http://vrm.ao2.it/>
11 <http://severncalaystudio.wordpress.com/bluebeard/>
12 <http://freestyle.sourceforge.net/index.php>

3) Under the 'Render' menu, open the 'Freestyle' sub-menu. Find the empty 'Freestyle Control Mode' box (label may appear as 'Control Mo'), and select 'Python Scripting Mode'. Then click on 'Add Freestyle Module', a default will appear.

4) Hit F12 to render.

[Noob Question: How to install Freestyle in Blender 2.61?]

Freestyle is being integrated into Google Summer of Code (GSoC). You can find experimental test builds... *i.e. Not Finished Yet...* at **GraphicAll.Org**¹³, search for "*GSoC Freestyle*".

Freestyle integration into Blender: Blog¹⁴

Freestyle integration into Blender: FAQ¹⁵

¹³ <http://graphicall.org/builds/index.php>

¹⁴ <http://freestyleintegration.wordpress.com/>

¹⁵ <http://freestyleintegration.wordpress.com/frequently-asked-questions-faq/>

94. Basic Animation

94.0.1. Note About Blender 3D: Noob to Pro/Basic Animation

I created this little note section to both warn learning users, and to give information to those who would make this wikibook better. The animation section in this book leaves a lot to be desired. It needs to go slower, and be more thorough. It is not finished (hence the accurately drawn "status images"). Here are a few things that need to be mentioned:

- What a key is
- How to manipulate keys after they are drawn
- How to delete keys (both IPO and Lattice)

I recommend that new users read the animation section in the blender documentation first. It is written for an older blender version, but it is more informative than this tutorial, and most of the information in there is still accurate. Here is a link: <http://www.blender.org/documentation/htmlI/p6390.html>

94.0.2. Frames and Keys

Animations are composed of multiple **frames**. If you pause an animation at any point, you'll be looking at a single frame in the whole of the animation. The most classic example of this is flipping pages quickly to produce an animation, as it has a different frame drawn on each page, and each page is displayed at a specific point of time.

A **key** is like a marker at a specific frame, which marks a particular thing (e.g. position, rotation, scale). If for instance, you have a character, then think of your key as telling Blender, "I want this character at this pose in this frame." To make your character stay in a certain position, you add a key or MARK it. By doing this, you're letting Blender know that this is an important pose. Now, if you move to a later frame and then adjust the character's position to another pose, then you can add another marker. What blender does, is it automatically makes your character move from the first position that you marked, to the second position that you marked again, in a very smooth manner. This, of course, depends on how far apart the keys are and if each key's content is positioned correctly.

94.0.3. Basic Animation Introduction

Animating, in principle, isn't that hard. Good animation is a very different story. These steps show you how to make a simple keyframe animation:

1. It is a good idea to switch into the animation screen using the screen selection drop-down on the main menu or by pressing CTRL + Left Arrow.

2. Go to the starting frame, using the arrow keys on your keyboard (Shift+Left Arrow goes to Start Frame, Shift+Right Arrow goes to End Frame, the Up and Down arrows Skip 10 Frames, the Right and Left arrows skip 1 Frame)
3. Go into the correct mode. To animate solid objects, go into Object Mode. To animate bones, go into Pose Mode (in this case, you don't have any right now).
4. Place the object or bone at the desired starting location and/or rotation.
5. Press the **i-key** on your keyboard while the mouse is in the 3D view and you will see a menu of the different properties which you may animate. Choose the most appropriate one, based on the properties you wish to change in the next keyframe. For example, if you wish to change the location, rotation *and* size of the object, select LocRotSize (now called LocRotScale). You have now made the first *keyframe*, which is a frame of animation which you have personally set up.
6. Now go to the frame of the next keyframe you wish to define, and put everything in the place you want for that frame. Remember, if you have a frame rate setting of 25 frames per second (fps) then if you want to make the next keyframe one second later in the animation, you need to go to frame 25 to make the keyframe. Then press the **i-key** again, and select the correct option again, depending on the changes you've made. (All in-between frames are automatically made to *interpolate*¹ between the two neighboring keyframes).
7. Repeat the previous step for each keyframe of the animation.
8. If you want more control over the transition between the keyframes, use the IPO-window (if you followed step 1, it should already be open on the right of the screen). Select the object you have keyframed and the IPO-window will display its animation curves (one curve for each of the properties you selected). You can select and edit the curves using the normal Blender controls. Use the curve menu for more options, such as making the curves actually curve instead of being straight lines.

For more information, see this [YouTube Tutorial](#)² or [The Blender Userguide](#)³

1 <http://en.wikipedia.org/wiki/Interpolation>

2 <http://www.youtube.com/watch?v=FAZFfLmbD2k>

3 <http://download.blender.org/documentation/htmlI/>

95. ScreenLayout

95.0.4. Intro

If you haven't done anything to the default blender startup, starting blender will produce the basic 3D window with a light, a camera, and cube, the buttons window, and the ever-present basic drop down tabs at the top of the screen known as the user preferences window. This is the default setup for modeling, but now that we are starting animation, we are soon going to be looking at a new(ish) set up. You'll see what I mean.

95.0.5. Setup

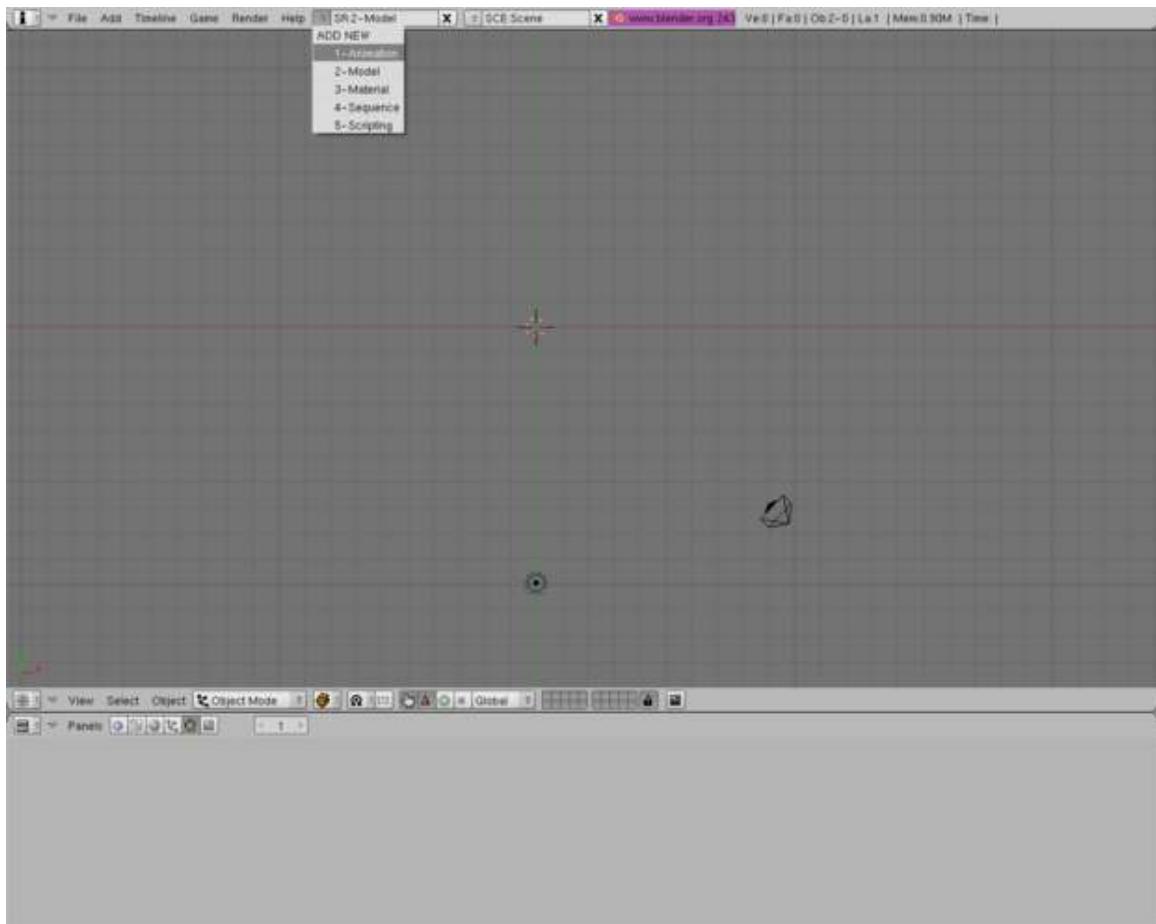


Figure 769 Changing to animation mode

Now the first thing we need to do is enter 'animation mode.' To do so, look at the top of the window, the 'user preferences window.' You should see file, add, timeline, game, render and help. What we want is the little drop down menu next to 'help' that you've probably never touched much less noticed. It should read something like: 'SR:2-model' with little buttons on both sides of the text. Click the button on the left side that looks like two little arrows to open a drop down menu. Click on 1-Animation.

95.0.6. About this setup

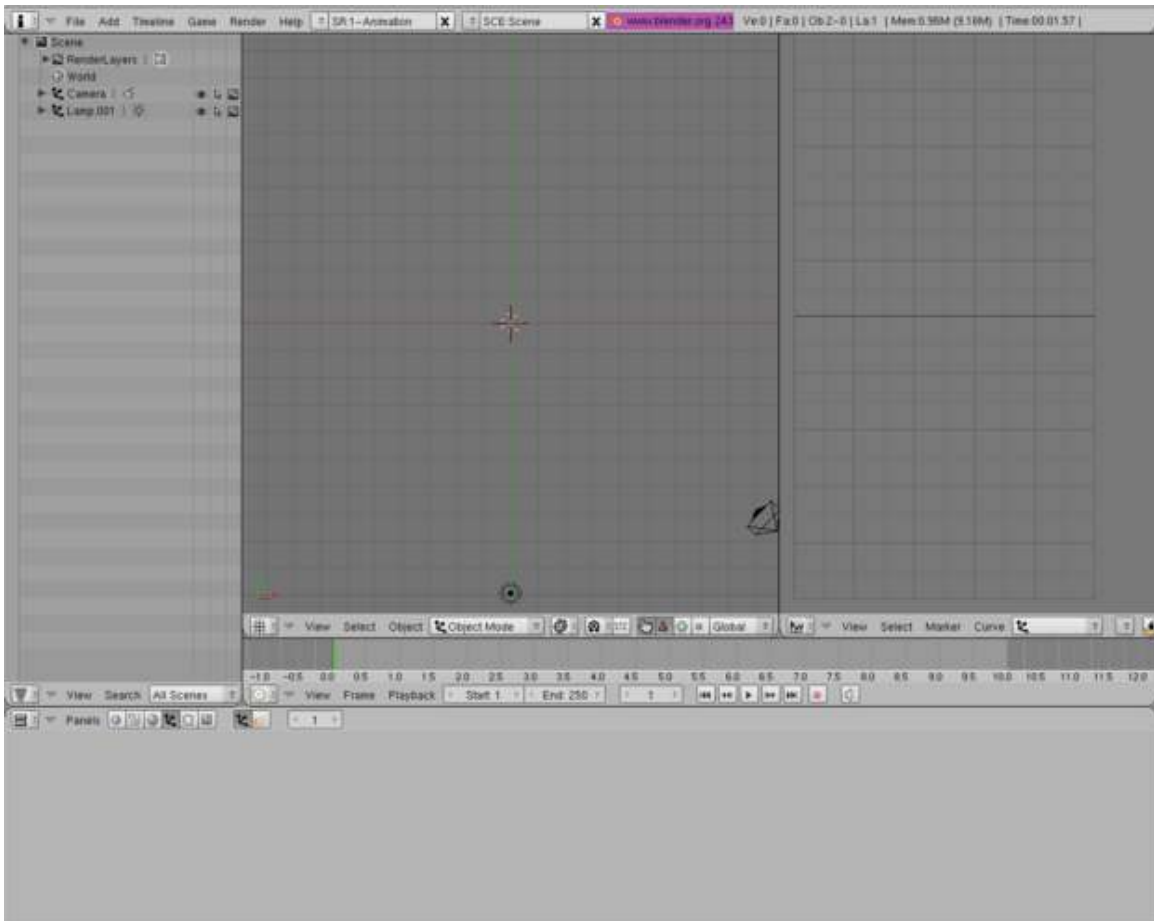


Figure 770 The Animation Mode

No, Blender didn't just go crazy, but as you can see, there are some changes that need to be explained. Blender has just opened a set of new windows, each with its own functions and commands. You can probably recognize three of them. The 3D view is still there, but it's been resized smaller to fit the new stuff. Our old friend the buttons window is still there at the bottom, and of course the user preferences window is still at the top. But enough of what's the same - you're wondering what the heck all the new stuff is for.

There are now three new windows in all. From top left going clockwise, the first is the Scene window. This window is used for finding objects and such located throughout your 3D environment. By clicking on any one of the objects listed here, its counterpart in the 3D viewscreen will be selected.

You can also select and tinker with your world settings and render layers from this window. The next window is the **IPO (InterPOLation)** curve window; it is the window that displays data concerning how your objects will actually move, and can begin to look very technical as we progress. Don't worry, it's normal. The final window is easier to miss. It is essentially a timeline, and is rather thin located between the 3D view/IPO windows and the buttons window. This is the Timeline window. For what we'll be doing for now, it will basically function as a kind of VCR. It is essentially the time codex associated with your animation. Clicking anywhere on the timeline will show you the animation as it is at that point, like fast-forwarding/rewinding then pausing. Like any good VCR the timeline window is equipped with the Play, Stop, Fast-Forward, and Rewind button set.

96. Lattice

96.1. What is a Lattice?

A Lattice is essentially a simple container that can be used to deform and manipulate a more complex mesh in a non-destructive manner (i.e. A lattice can be used to seriously deform a mesh then, if the lattice is later removed, the mesh can automatically return to its original shape).

96.2. How to add and use a Lattice

A Lattice is added to the scene in the same way other objects are added. Either:

1. **Shift-A** over the 3D window and choose **Lattice** from the pop-up menu, or
2. Press **Spacebar** over the 3D window and choose **Lattice** from the pop-up menu, or
3. LMB **ADD** from the 3D window menu and choose **Lattice** from the drop-down menu

The default Lattice looks just like a cube when first added except that it is just one Blender Unit (BU) wide whereas a mesh cube is 2 BU wide. When the Lattice is added, the window remains in **Object Mode** and the Lattice can be moved, resized and rotated like any other Blender Object.

On its own, a Lattice serves no purpose whatsoever since it can't be seen in a rendered image. Its only use is to manipulate another object and so, to be useful, we need to associate another object with it. The current method for doing this is by applying a "**Lattice Modifier**" to a Blender object.

The basic workflow is:

- Add a mesh (cube, cylinder, cone, sphere, etc.)
- Add a Lattice
- Select the mesh
- Press F9 and go to Modifiers Panel(buttons window). In the modifiers tab Press **Add Modifier > Lattice**
- In the Ob: Panel, enter the name of the Lattice (The default name is Lattice, Lattice.001, Lattice.002, etc. - or you can give it a useful name that you'll remember later)

NOTE:In older versions of Blender, a lattice was applied using a parent method - select the mesh, then select the Lattice, then press Ctrl-P and choose "Lattice Deform". While this method is still available, it does not offer the full functionality of the newer Lattice Modifier. If you use the parent method, you will need to press the "Make Real" button in the modifiers palette to enable the modifier functionalities. Using the method, the Lattice becomes a parent and a modifier. As a parent, it will act on the mesh in Object Mode whereas a true modifier lattice only influences the mesh when altered in Edit Mode.



Figure 771

Adding a Lattice Modifier to an object

Now, we can change the Lattice in **Edit Mode** and any changes we make to it will affect the mesh.

Note: Applying the Modifier Correctly

When adding the modifier to your object, you must enter exactly the same name as your lattice. The default is **Lattice** with a capital "L" but if you've changed it or have more than one Lattice, then you will need to enter the new name. When you enter a valid Lattice name, the name stays visible in the Ob: box. If your entry disappears then you've entered the name incorrectly.

One simple way to get the right name is to select the Lattice, go to F9, Link and Materials panel and where it says Ob:Lattice or Ob:Lattice.001 etc., move the mouse over this field and press **Ctrl-C** (*don't click on it, just hover over it*). This copies the Lattice name. Then select your object, go to the Lattice Modifier panel, hover the mouse over the Ob: field and press **Ctrl-V** to paste the name in. Now it should stay there and your Lattice should work in Edit Mode.

Noob note: I recommend the method with **Ctrl-P** for animation.

96.2.1. Basic Exercise:

1. Start with a new Blender scene, delete the default cube and add a UVsphere (Spacebar>Add>Mesh>UVsphere). Accept the default 32 Segments and Rings. You could use

any of the mesh shapes but because the sphere is heavily subdivided (*made up of lots of edges and faces*); you will get a better idea of what the Lattice is doing.

2. Tab back into Object Mode then add a Lattice. It is generally wise to resize the Lattice so that it surrounds the mesh it will be deforming. So, press SKEY and enlarge the lattice to 2 BU wide.
3. Select (RMB) the Sphere and add a Lattice Modifier to it. Make sure that you type the exact Lattice name correctly into the box. Nothing appears to happen but that's fine.
4. Now select your Lattice, TAB into Edit Mode, select just one control point (a control point looks like a vertex) on the Lattice and move it around. You will see the sphere mesh stretching and squashing relative to that control point. Move each Lattice control point one at a time and see just how far you can deform the mesh. Zoom in your 3D window if you need a closer look.
5. The control points of the Lattice can be moved, scaled and rotated in the usual ways. Try selecting a few of all the control points and scaling them (SKEY) or Rotating them (RKEY) and watch the mesh follow along.
6. Now, exit Edit Mode (TAB) and select the sphere in Object Mode. Go to the modifiers panel and press the big "X" button to remove the modifier. Despite all that deforming, the sphere immediately returns to its original size and shape. Nothing was really changed in the sphere's mesh data. Immediately re-apply the modifier as before and see the Lattice immediately apply its own deformation to the sphere again.

96.3. Getting more involved with Lattice

The default Lattice is two control points high, two wide and two deep, i.e., it is two control points wide in each direction (these are referred to as the U, V and W directions). However, we can change the number of control points in one, two or all directions. This is done by selecting the Lattice, going to the Lattice Panel (F9) and changing the values in the U, V & W buttons. If you decrease one figure to a value of 1, the Lattice will become two-dimensional (planar). Decreasing two values to 1 will change the Lattice to a line (one dimension). This can be useful, especially if the remaining value is increased, but is not the most common usage of Lattice.



Figure 772

UVW settings for Lattice

NOTE: The **"Make Regular"** button will set the UVW control points of an unscaled Lattice to be exactly one Blender Unit apart in each direction. The **"Outside"** button effectively removes all the internal control points which are added when the UVW settings are higher than 2.

In most situations you will want to increase some or all the values as this gives you more control over complex deformations, much like a subdivided mesh. How much you increase the UVW directions depends entirely on how much detailed control you need over deformations. As you've already seen basic squash, stretch, shear and simple deforms can easily be achieved with the default 2,2,2 Lattice but by increasing the UVW values a little, a whole new range of deform possibilities becomes available.

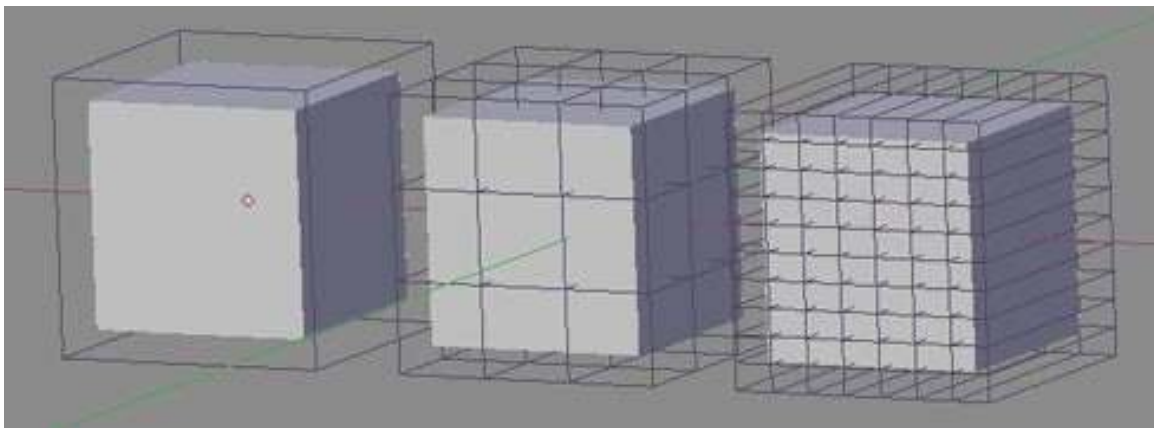


Figure 773

Three Lattices with different UVW settings

One thing to understand, however, is that a Lattice cannot bend the individual edges of a mesh (the lines connecting any two vertices) so the mesh must contain enough edges in order to apply complex lattice deformations to it. These edges must be genuine edges and not the virtual edges created by a subsurf modifier (Correction: Yes, they *can* be edges generated by a subsurf modifier, provided the subsurf comes above the lattice in the modifier stack for the object being deformed; use the up/down-arrow buttons next to each modifier to rearrange them as necessary). Edges can also be added to a mesh using a variety of tools including subdivide, knife and loop cuts (some info on these http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Die_Easy_2¹).

New Note: Use the K key to open a menu for the knife, loop cuts, etc. Knives are used to add a vertex to each line you left click and drag the cursor over, and loop cuts allow you to make multiple cuts into your active object.

About loop cuts: Ctrl + R also is a direct shortcut to a loop cut. Furthermore, loop cuts' main uses are for subsurf modeling. In order to accomplish a hard edge in a subsurfed object (for example the edges and corners of a wooden table or the bumpers of a car) you must apply loop cuts to the faces surrounding the edge and place them very close to them (and parallel). Experiment with a simple subsurfed cube and you'll see. Keep in mind this is why when modeling it is highly recommended to use quads. A loop cut will only cut a quad face, or every face connected with it in one direction, but only quads. Another main reason why it is recommended to model with quad faces is regarding animation but I suppose other tutorials will deal with that later. More information: http://wiki.blender.org/index.php/Manual/Edge_and_Face_Tools²

1 http://en.wikibooks.org/wiki/http%3A%2F%2Fen.wikibooks.org%2Fwiki%2FBlender_3D%3A_Noob_to_Pro%2FDie_Easy_2

2 http://wiki.blender.org/index.php/Manual/Edge_and_Face_Tools

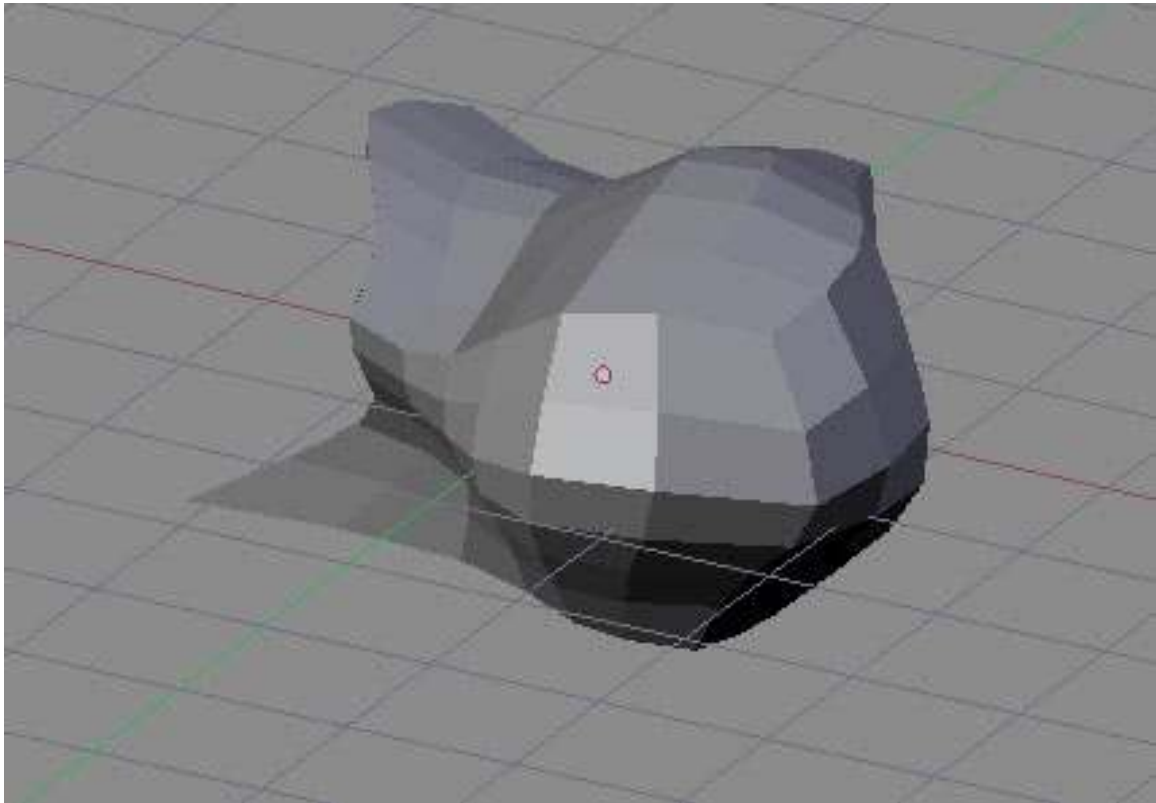


Figure 774

A simple subdivided cube after a complex lattice deformation

96.3.1. Intermediate Exercise

For this exercise, start over with a new scene and repeat Steps 1 - 3 in the Basic Exercise above. Don't go into Edit Mode yet.

Basic Exercise:

1. Start with a new Blender scene, delete the default cube and add a UVsphere (Spacebar>Add>Mesh>UVsphere). Accept the default 32 Segments and Rings. You could use any of the mesh shapes but because the sphere is heavily subdivided (made up of lots of edges and faces); you will get a better idea of what the Lattice is doing.
2. Tab back into Object Mode then add a Lattice. It is generally wise to resize the Lattice so that it surrounds the mesh it will be deforming. So, press SKEY and enlarge the lattice to 2 BU wide.
3. Select (RMB) the Sphere and add a Lattice Modifier to it. Make sure that you type the exact Lattice name correctly into the box. Nothing appears to happen but that's fine.
 - In Object Mode, select the Lattice then press F9 and go to the Lattice Panel. Increase all the UVW values to 4. You will immediately see the change displayed in the 3D window.
 - Tab into Edit Mode and play with the control points again. The first thing you might notice is that the corner control points have less influence now than before. This is because the effect is

proportional to the distance of each point from the mesh. Move some points in the middle of a side face and you'll see the effect is significantly greater.

- Try selecting groups of control points and pulling them away from the Lattice to stretch lumps and bumps out from the mesh. If you are familiar with the proportional edit tool, try that too - it works on Lattices just like it does on meshes.

Try doing the same exercise but increasing values in only the *W* direction. (*U* divides in the *X* direction, *V* divides the *Y* direction and *W* divides the *Z* direction). So, set the UVW to 2,2,4 and play with the control points again in Edit Mode. Select the two rows of control points around the centre of the Lattice and scale them up (SKEY) then Scale them down along *Z*-axis only (Skey>ZKey). Try other transforms constrained on different axes for interesting, controlled results.

There are three different options for how the Lattice Modifier affects each UVW direction. These are Linear, Cardinal and B-Spline. All I can say is that B-Spline is the default and to find out what the difference is, press the buttons and see.

96.4. Examples

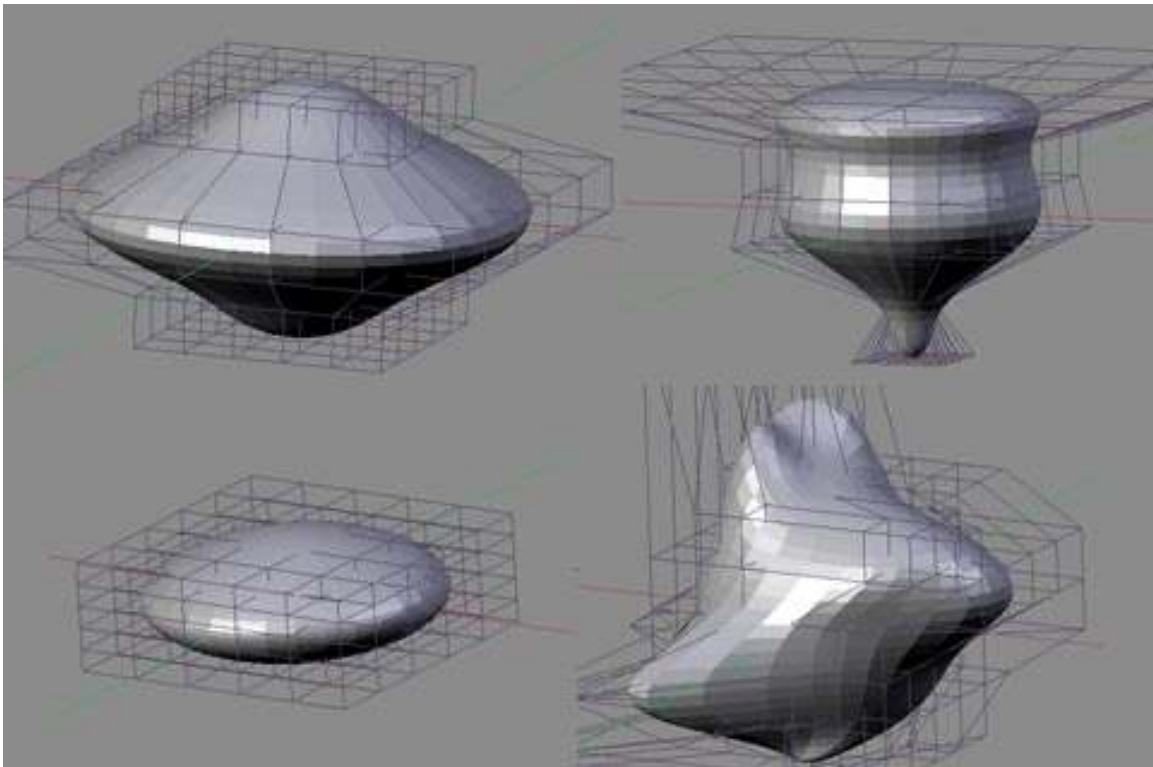


Figure 775

Some examples of a Lattice deforming a standard UV Sphere mesh (not smoothed)

96.5. Making it stick

To keep the mesh deformed permanently, you can select it and press the "APPLY" button in the Lattice Modifier panel. This "bakes" the mesh in the deformed position and disconnects it from the Lattice. The Lattice can then be deleted without the mesh returning to its original shape. This is useful if you are using the Lattice as a modelling tool rather than an animating tool.

Also, instead of deleting the Lattice, you can use it to immediately modify another mesh. Simply move the Lattice over the new mesh in Object Mode then apply the Lattice Modifier to the new mesh. If the Lattice is already in a deformed state, the mesh will immediately be deformed too. Press Apply again to "bake" that mesh and keep re-using the already deformed Lattice on as many other meshes as you wish for matching results.

[Noob Note: I found it interesting that you can even apply the lattice deformation to the original object again once baked, and it will deform even further.]

96.6. Animating a Lattice Modifier

Lattice animation uses a workflow almost identical to the RVK (Relative Vertex Key) workflow from Blender 2.37 and earlier. *You don't need to know this but some people might find the information useful.*

With your Lattice Modifier already added to your object, select the Lattice and press **I-Key** to tell Blender you wish to animate the Lattice. Choose "**Lattice**" from the pop-up list. This sets the basis key (undeformed state) for the Lattice.

Adding the basis key

In the Editing panel (F9), press the "**Relative**" button.



Figure 776

Set the keys to "Relative"

NOTE: The Slurph setting determines the delay, in frames, for how long it will take to morph from their former state to the one applied by the new lattice.

To set your first deform key, press **I-Key** and then "Lattice" again then enter **Edit Mode**. Deform the Lattice by scaling or moving control points then Tab back into **Object Mode**. If you open an **Action Window** now, you will see your first key, "Key 1", added to the list. If you press the small arrow at the top of the list, it will display the **slider** for that key.

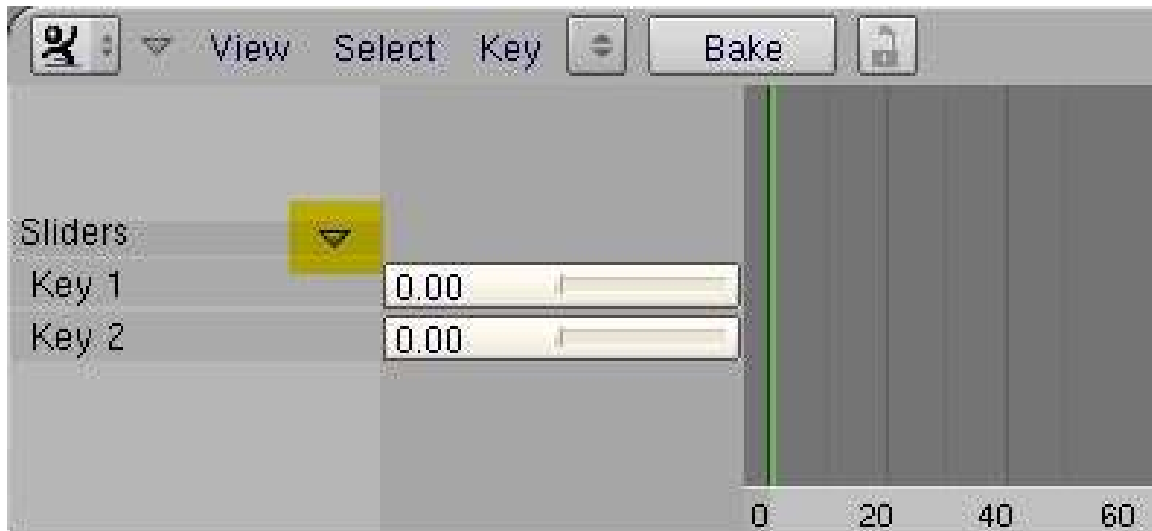


Figure 777

Lattice key sliders in the Action Window [Noob note for blender 2.48a] In the action window, select "ShapeKey editor" in the dropdown "editing modes for this editor" (in the panel header)

To add more Lattice keys, for a variety of deform shapes, repeat the process: I-Key, Tab to Edit Mode, move control points, Tab back to Object Mode. Each time you exit edit mode, a new key will be added to the list in the Action Window.

Animating the Lattice uses the same process as animating Shape Keys for meshes.

If you want the object to begin undeformed, then set each key slider to zero on the first frame of the animation. Move through the frames, setting the sliders as you go to deform the Lattice as desired. You will often find you'll need to set a key before and after each deform key in order to control the rate at which deformations take place in the animation.



Figure 778

Setting keys in an animation

Noob note: I'm doing everything as described and get no keys in action window when exiting Edit mode... Any suggestions?

Noob note; Is it possible to create a keyframe using the object's mesh rather than applying a lattice?

Noobie note: What to press after I-Key? I'm not getting any keys in the action window, help please.

Another Noob: No keys here. I got something changing to IPO Curve Editor and choosing Shape in IPO Type

Noob note; Select Action Editor header->Shape Key Editor. Make sure you also selected Buttons Window->Editing (F9)->Relative Keys

Noob Tip: I am seeing a possible bug in 2.49 in which the key sliders will not appear on the Action Editor unless there is at least one window set to Action Editor when the keys are created. To reveal key sliders for keys that have already been created, save the file, re-open it, set one window to "Action Editor" set the window sub-type to "Shape Key Editor" then switch the Lattice "Relative" setting on and off (putting the cursor over the Action Editor window and hitting HOMEKEY will help locate the sliders). Or, by preference, make sure at least *one* window is set to "Action Editor" before creating the keys.

Noob note: I couldn't see the sliders either; then I scrolled up and there they were, off the visible portion of the window. However, I had by then change the window type a few times, so I can't rule out that that did something first.

96.7. Let the fun begin!

Now you have your object safely locked away inside your Lattice, you can still animate the object itself, inside the Lattice!

96.7.1. Why would I animate the object too?

Take the case of the classic cartoon eyeball which is taller than it is wide. If you just stretched the sphere object itself, instead of using a Lattice to deform it, then you couldn't properly rotate the eyeballs to look up and down because the whole "egg-shaped" eyeball would rotate and end up lying on its front or back. The eyes would literally pop-out of the head.

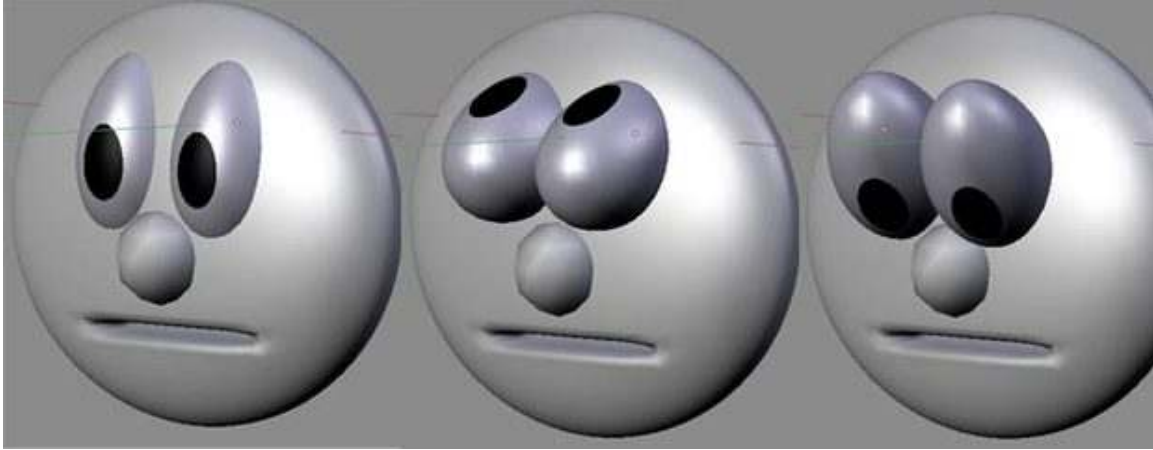


Figure 779

Cartoon eyes: The stretched spheres don't rotate properly.

If, however, you use a Lattice to deform the basic sphere to make it into a tall "egg" shape, you can still select the eyeball itself and animate it within the Lattice. Now when you rotate the eyeball, it will look up and down but still maintain its deformed shape within the head.

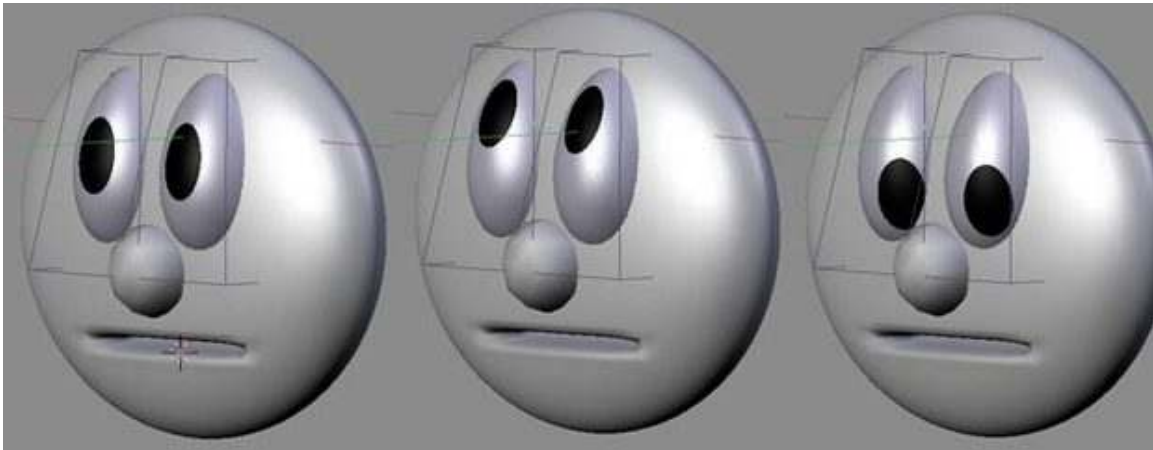


Figure 780

Cartoon eyes: Spheres deformed with Lattice can still be rotated with good results.

97. Bounce

97.1. How to make a ball bounce convincingly!

This tutorial assumes basic Blender awareness and some knowledge of using Blender Lattices as well as keyframes basics regarding the Blender Panels. Knowledge of IPO curves will prove useful later as we progress to more advanced techniques.

97.2. How hard can it be to move a ball?

It would be easy to just put a sphere on the screen and animate it to move up and down but in all honesty, it would not look like a bouncing ball. It would not be convincing in any way.

To be believable, the ball must use some of the most fundamental principles of good animation. In particular, the ball must **squash** and **stretch** and change **speed** as it falls, hits the floor, bounces and rises ready to fall again. With a little effort we can make that boring sphere look alive!

97.3. Why use a lattice?

It is possible to make a simple bouncing ball animation without using a Lattice object but with the Lattice we can do more than just bounce the ball. For example, once we have a ball that bounces how we'd like it, we can later add rotation to the ball so it spins through the air and bounces then, as the bounces decrease, the ball can roll to a halt. Doing this without lattice would be a far more complex exercise as a rotating squashed and stretched sphere would look like it was wobbling in space (*like having the flat part of a flat tire spin around the tire instead of staying on the bottom, even while the wheel is turning*).

97.4. Let's begin then

To begin, add a UVSphere (SPACE *add>mesh>UVSphere*) in **Top View** (NUM7). It is important to add the objects in top view to keep their **Z-Axis** aligned upwards. Set Smooth and Sub-surf on the sphere if desired. Tab back into **Object Mode** then immediately add a Lattice (SPACE *add>Lattice*). Resize (SKEY) it so it closely surrounds the sphere. Make the edges of the lattice align with the circumference of the sphere as this will make animation easier.

(By default the Lattice Object Name is "Lattice". It is good to give all your objects unique and sensible names, especially if you are planning a complex scene. So you might call the Sphere

"Red_Ball" and the Lattice "Red_Ball_Latt" or something similar for easy reference and recognition later.)

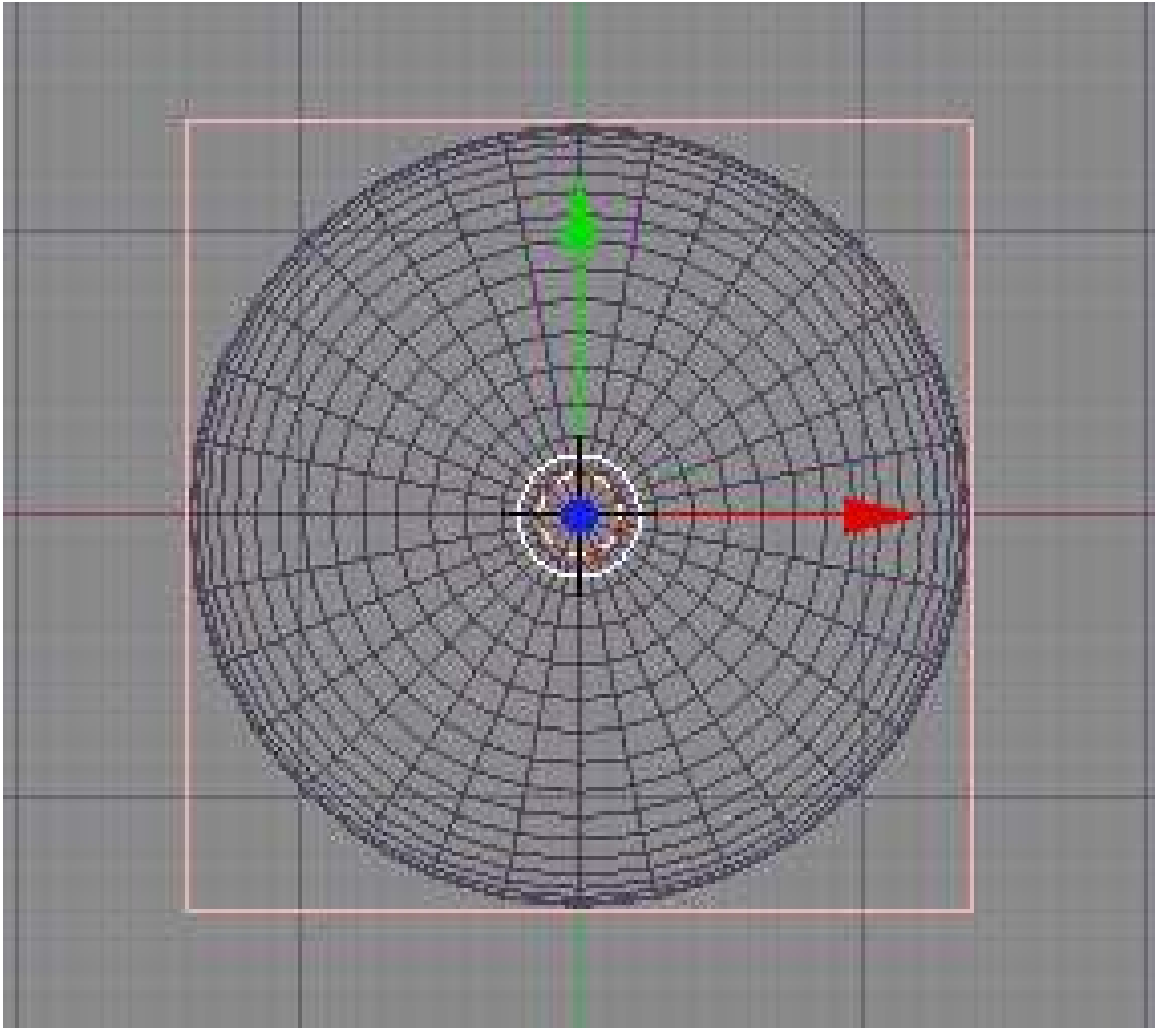


Figure 781

Top view of UV-Sphere inside selected Lattice

97.4.1. Make the lattice deform the sphere

Prior to Blender 2.4, objects were parented to Lattices with a Lattice Deform option. In version 2.4+ you can select the object (the sphere) and apply a Lattice Modifier to it (Ob: Lattice). However, if you use this new method then the ball will not move where the Lattice moves and will not squash when the Lattice squashes and we want it to do both of these things.

Luckily you can still use the good old **Parent option** then use the Modifier palette to make the Lattice Deform "Real" . So select the sphere and THEN SHIFT SELECT the Lattice then **CTRL-P** to make parent and choose the "**Lattice Deform**" option. If you now select just the sphere, press **F9** and look in the Modifiers palette, you'll see a listing for "**Lattice parent deform**" with a button for

"**Make Real**". You can either ignore this button or press it. It really won't matter for now. If we wanted to do some weird things to our ball, like kicking it or deforming it in other complicated ways, then we would have to use the "real" Lattice modifier option.

So, this may not be the purist approach to using a Lattice but it is a convenient way to achieve the results we're after. Basically, it works and is reasonably intuitive.



Figure 782

Lattice Modifier screen using parenting option.



Figure 783

If you press "Make Real" you enable the full Lattice Modifier

You can easily check if the Lattice is working the way we want it to by selecting the Lattice and resizing, moving or rotating it. **The ball should do everything the Lattice does.** I don't wish this to be a full lesson in Lattices so if you don't understand any of the above, or you're certain your Lattice is not working as it should, then please learn a bit about lattices¹ before proceeding.

97.4.2. Time to animate!

For our basic animation, we'll use just 23 frames starting with the ball high then falling, squashing, bouncing and ending up back where it started. When this is played back in the 3D window, it will loop and we'll see the ball bounce forever in one place.

This part is easier done in "SCR:1 - Animation" Viewport configuration. It is also useful to have multiple orthographic views open in the viewport by right clicking on the resize viewport arrow and choosing "split area" then making a top view, side, etc...

Go to **Front View** (Num1) Press F10 and set your start (Sta) frame to Frame 1 and your end frame to Frame 23. Then make sure you're on **Frame 1** (Shift+Left Arrow) to start animating. Note that we are **animating the Lattice**, not the ball. Everything we do to the lattice will directly affect the ball too.

Note: all the modifications to the lattice must be done in 'Object' mode. If not, they will affect all the frames.

Select the Lattice (**NOT** the ball) then do the following on the corresponding frames:

¹ Chapter 96 on page 821

- **FRAME 1:** Press IKEY and set a **key** for LocRotScale - the ball's start position (high).
- **FRAME 11:** Move the Lattice to ground level and **key** LocRotScale again (*keep the fall distance small for now*).
- **FRAME 13:** Leave Lattice in same position as frame 11 and **key** LocRotScale again.
- **FRAME 23:** Move Lattice back to start position (use Num (N) Panel if you need to) and **key** LocRotScale again.

(Noob note- This may seem obvious, but move the lattice BEFORE you insert the keyframe.)

These four frames give a basic up and down animation with a pause at the bottom. We can add squash on Frame 12 to finish off.

- **FRAME 12:** Place the cursor at the base of the Lattice then set the Rotation/Scaling Pivot to "3D Cursor"

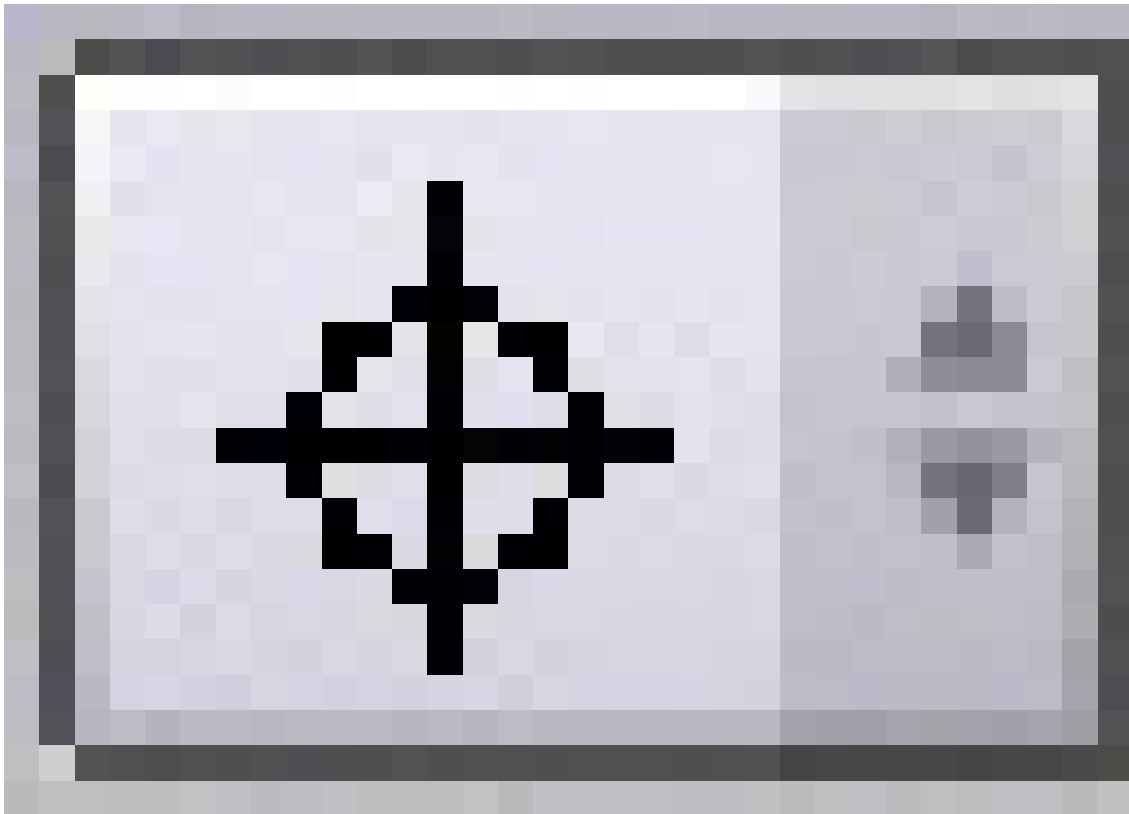


Figure 784

Scale the Lattice down along the Z-Axis **SKEY-ZKEY** and see the ball squash. **IKey** LocRotScale again.

When you have set all five keys, you can cycle through the frames and you should see the following result on the Key frames we just set.

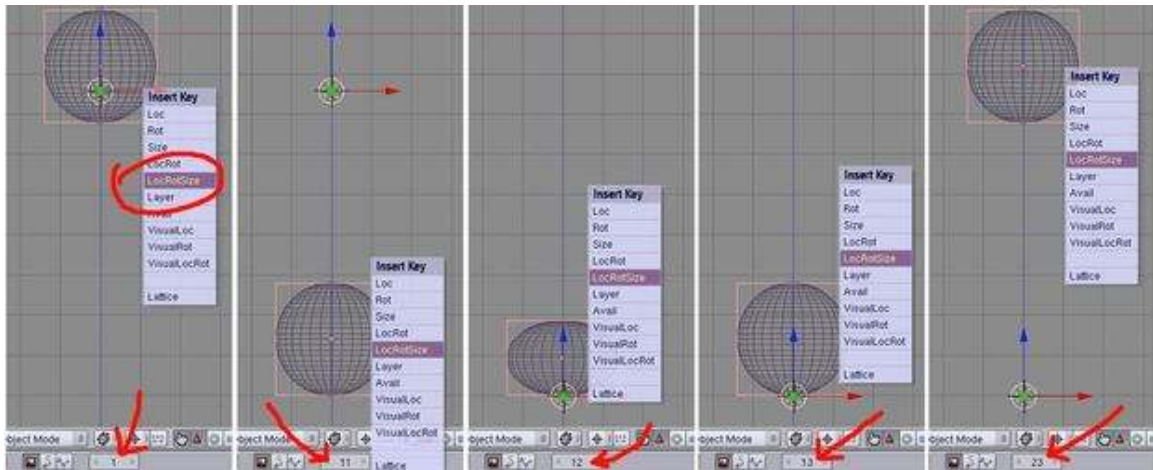


Figure 785

Screenshots of the 3D window, front view, on each of our five key frames.

If you press **Alt-A** now in the **3D window**, you should see the ball bounce. Even though the squash happens on just one frame, it still reads properly in our minds and is a vast improvement over an animation with no squash at all. (Press **RMB** or **ESC** to stop the animation playback).

If you have a camera and lighting suitably set up you could render the animation out and use Blender's "PLAY" button to loop the rendered animation. See the bottom of the page for an OpenGL render made from the steps above. Add material to the ball and a ground plane for a more interesting result.

97.4.3. Looking at the IPO Curves

If you want the ball to bounce three times then you could repeat the above instructions three times over - if you have the patience - but Blender has a better, easier way.

Select the Lattice then open up an IPO (**InterPOLation**) window and zoom into the group of keys. If you select LocZ and SizeZ from the channel list on the right, you should see something like this (I've added labels to the screenshot):

Note the the IPO Type here is "Object". Purists may be disturbed or perplexed by this but if you remember, we're not really using our Lattice in the purist, Lattice Modifier form. Don't worry about it though. It works.

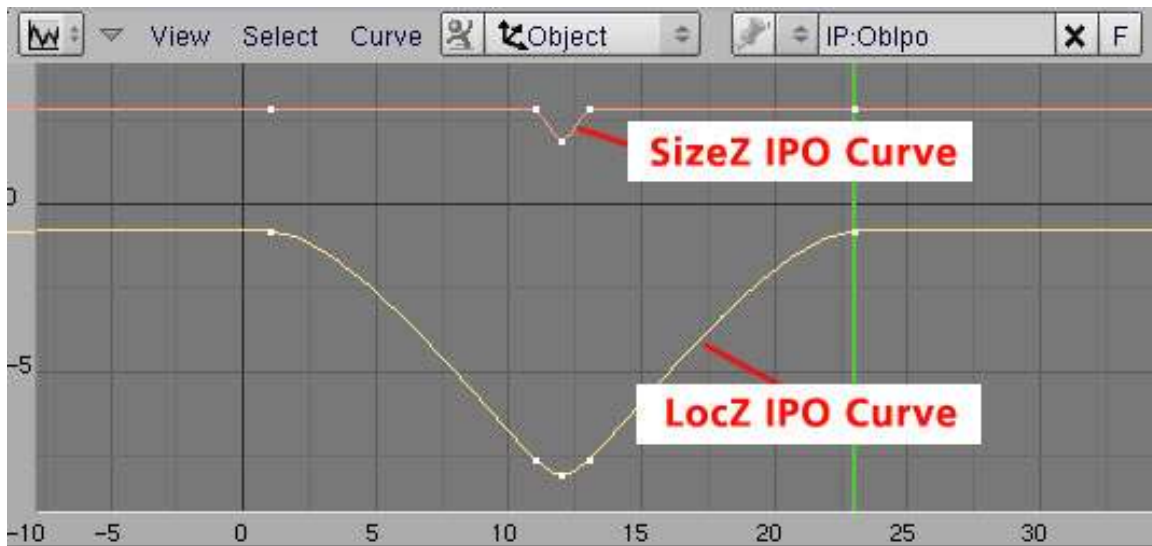


Figure 786

IPO curve for Lattice, showing LocZ and SizeZ channels.

This shows the relative position on the Z-Axis and size in the Z-Axis of the Lattice. You can see that the Size remains constant until frame 11, then reduces, then goes back and remains constant until frame 23. Similarly, the LocZ curve shows how the ball starts high, falls then stops before rising again. The rounded bottom of the LocZ curve occurs because the IPO curve maps the centre point of the Lattice and when we squash it, the centre point lowers slightly.

Of special interest is that by default, Blender uses Bézier curves for IPOs and this gives us rounded tops on the curve. This means the ball starts moving slowly downward then moves faster until it reaches the bottom. Also, as it nears the top, it slows down again. This is good because this is exactly what happens in reality when a ball rises and falls (*It could be better - but it's a good start*).

While in the IPO window, select both curves then press **KKEY** and you'll see each key frame indicated by a vertical line. (Note: You can also press *View->Show Keys* if you can't remember this shortcut - this has helped me in the past.)

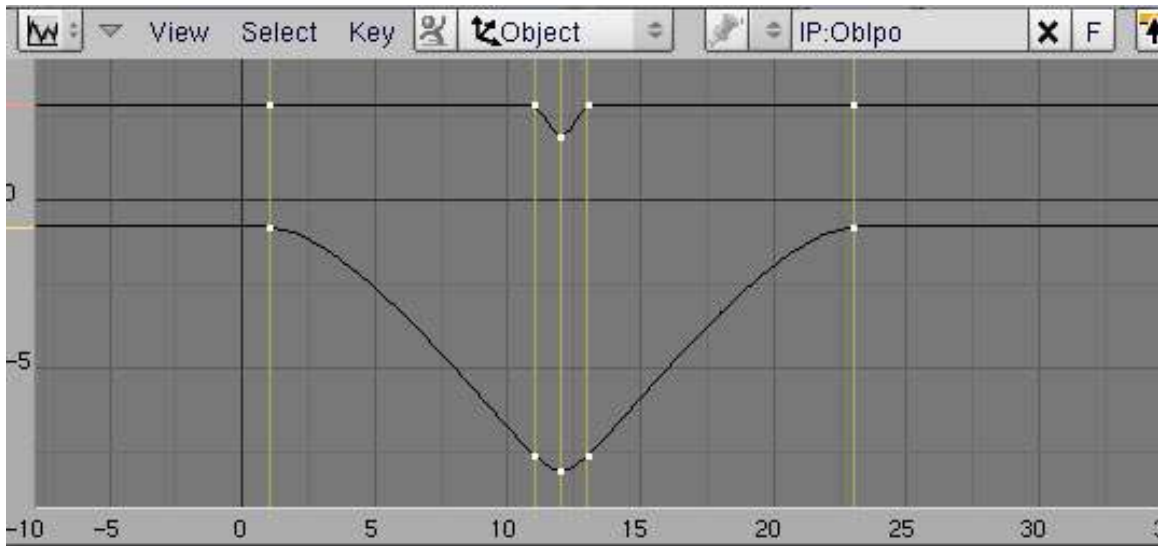


Figure 787

IPO window set to show key frames



Figure 788 Bouncing ball animated GIF

Press **AKEY** to select all key frames then duplicate (Shift-D) them. The curves will seem to deform but when you drag the frames to the right, they will sort themselves out. Position them so the "first" frame is at Frame 24 and the new end frame is now Frame 46. Reset the End Frame in the Scene (F10) window then watch the animation again. It's not bad. You can repeat this process for more bounces. (To have it bounce forever throughout a long animation, ignore the above procedure and just set the "Extend Mode" to "Cyclic" in the Curve menu. You'll have to adjust the curve a little on the first and last frame but you only do it once and Blender repeats the change for all other cycles.)

It's not bad for a start but there's a lot more to do to this ball before applying for that animation job.

97.5. Saving The Animation

How do you tell Blender where to save the animated frames? The answer is in the “Output” mini-window that you see in the Render settings (F10). Near the top of this are two editable text fields, the first of which is initially set on my Linux system to “/tmp/” and the second to “//backbuf”. The first one is the output pathname or directory prefix used to save the generated frames as JPEG files when you press the “ANIM” button. If it has no “#” characters in it, then it is used as a directory prefix; thus, the default will be to save the animation frames as /tmp/0001.jpg, /tmp/0002.jpg etc. Alternative, if you put in a filename spec with “#” characters in it, the hashes will be replaced with the frame number. Thus, if I set the field to “/home/ldo/Documents/BlenderTuts/ball#####.jpeg”, then the frames will be saved to that /home/ldo/Documents/BlenderTuts directory and be named ball0001.jpeg, ball0002.jpeg etc.

97.6. So what next?

In real life and in a typical cartoon, balls don't just bounce forever in one spot. They move along and slowly lose their bounce before rolling to a halt. And they have more character than our bouncing ball.

Things to consider next:

- Perfecting the squash - maintaining volume
- Adding stretch
- Look at acceleration and deceleration
- Rotate the ball as it bounces
- Reduce the bounce height over time
- Move the ball around as it bounces

See this [ANIMATION LINK²](#) (580kb, AVI-DivX) for an idea of how you can achieve interesting results with just a little more effort. but its k? k

97.7. Links

Principles of bouncing ball animation: <http://www.idleworm.com/how/anm/01b/bball.shtml>

Blender 2.49B bouncing ball animation: http://www.youtube.com/watch?v=BEY_6dFdUJY

Blender 2.49B vector blur animation, to add realism to your animations:
<http://www.youtube.com/watch?v=qY4WcNgEXv8>

² <http://members.westnet.com.au/andydolphin/blender/bounce.avi>

98. Creating Basic Water animation

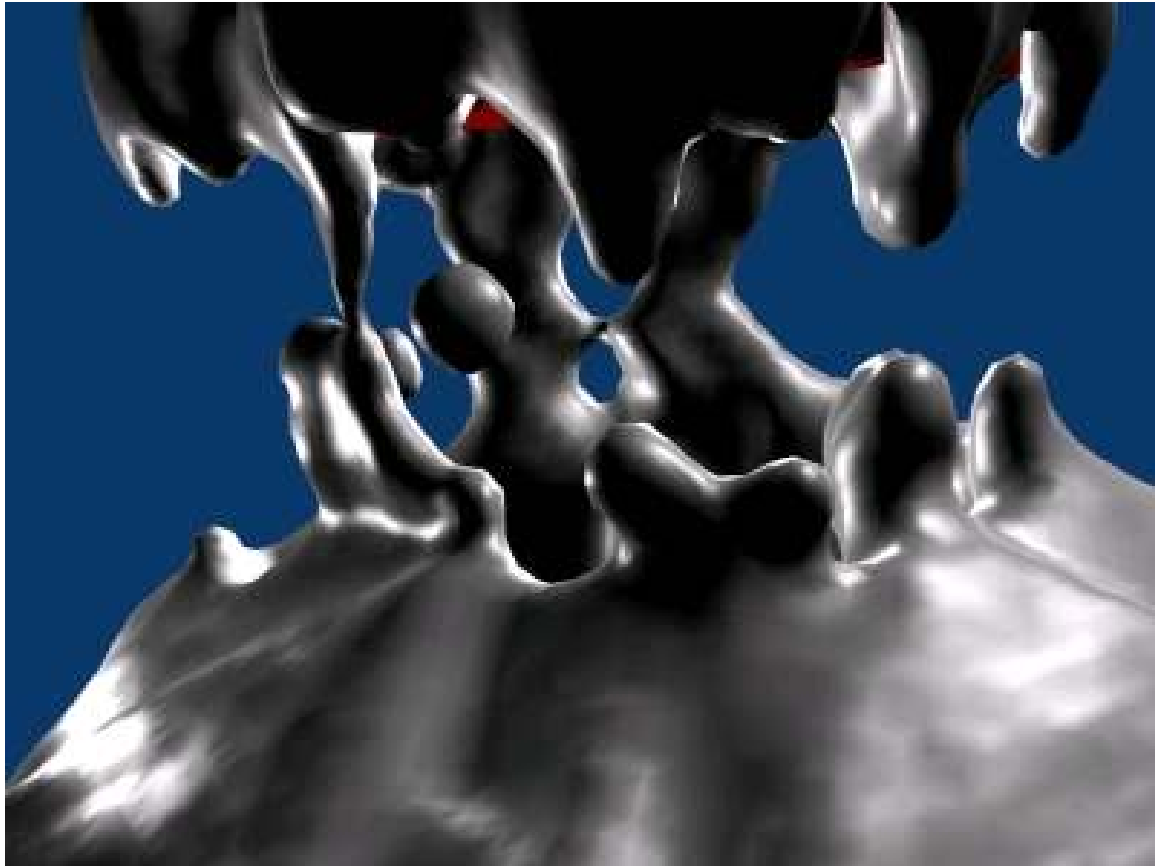


Figure 789 A low resolution, high viscosity fluid test

Important: Always make the domain after you have already made the fluid and everything else. Subsurf the object that you are currently working on before you create any other object except the domain.

98.1. Water and Other Fluids

Water is without a doubt one of the most important compounds in our lives; it covers about 70% of the earth and is therefore incredibly important in quite a few blender animations. Wouldn't it be great if we could get an accurate physical representation of this liquid in blender? We can, using a tool called Fluid Simulation. This tool looks unnervingly complex at first glance, but this tutorial should

clear it up for you. At least to a basic level. *(Sorry, but I'm not the best with this feature. If anybody sees any missing information that you can help with, please don't be shy to contribute!)*

98.2. The Domain

One can imagine how much time it would take Blender3D to think about everything in the infinite space of a 3d world in terms of fluid objects and deflection, so we obviously need to cut down on that size. Create a fairly large cube. This will eventually be set as the volume in which all of the fluid simulation occurs. Don't make it too large, but not too small either. With this cube selected, press "F7" to get to the physics screen. Find the tab that says "Fluid", and click the "Fluid" button to enable the function. Of the six new buttons that appear, click "Domain". All the fluid physics will be calculated inside this cube. Also in that tab, you'll see three buttons, "Std" (which will be pressed by default), "Ad", and "Bn". Each one opens a different set of values. These will be explained as they become important.

Noob note:

1. The example animated GIF below uses a domain which consists of the standard cube scaled 10 times. RMB, SKEY, 10, ENTERKEY.
2. The fluid can be set to be smaller than this. OBJECTMODE, add, mesh, cube, SKEY, 10, ENTERKEY. Reduce this second cube by half. EDITMODE, deselect all(AKEY), border select(BKEY) top vertices, pull down the vertices to half way.

98.3. The Fluid

As a basic start up for your experiments with fluid simulation, I am going to take you through a small demo in which we drop an object into a pool of water, creating a splash. To do this, we need a fluid object and an obstacle object. For the fluid, create another cube, scaled down to cover the bottom of your domain. Enable this object in fluid simulation also and select "fluid". When we start to bake the fluids, this will be set up as your liquid. Make sure you bake this animation COMPLETELY before you add the obstacle. (Don't bake it yet, this will be better explained in the Baking Fluids section)

Let us go over what you have now. You have your starting cube, with the "Fluid Simulation" and "Domain" buttons depressed, and will be referred to as "Domain". Inside this is the cube that covers the bottom of your Domain, which is the liquid. The liquid will be a different object with the "Fluid Simulation" and the "Fluid" button depressed. Create a third object which we will call "Obstacle". This will be the object dropped into the water, and will be described further in the explanation.

98.4. The Obstacle

<http://img153.imageshack.us/img153/240/blenderfluidtut1.jpg>

Now, create the object you want to drop into the water. Give it an IPO/Animation to drop into the water, and enable it as--you guessed it--an obstacle (click the "fluid" button, click "obstacle").

(User Note: We were just told to bake the animation "COMPLETELY" "before" adding the obstacle, but the baking instructions come in the next section. Are we supposed to bake the animation and then add the obstacle, as it says, or add the obstacle and THEN bake the animation as the sequence of instructions indicates?? *Could anyone out there clean this up a bit?*)

(User Note: He said when you bake it its important to bake the fluid before the obstacle, you are not supposed to bake it until Baking Fluids Step.)

Set up the animation in the front/side views, not the top view. The z-axis is where gravity works. No baking required at this stage. othman

98.5. Baking Fluids

Now we get to see what it all looks like. Before we do anything too fancy, however, go to the render buttons (F10) and specify the total number of frames to simulate the fluid in. Most projects don't need the whole 250 frames. Select the domain object, and find the big button that says "BAKE" under its fluid simulation panel. Press it, click the "continue" part of what comes up, and watch your masterpiece evolve. Be patient, blender has to render this simulation just like it would render the final product. The frame count of this rendering replaces the bar at the top of the screen where it says, "www.blender.org" with the blender logo. If at any time you feel that you want to abort this process, press "Esc". Once the bake is finished, open a timeline window and press play. There, you should be able to view the full simulation.

(note: Make sure the bake data is being sent somewhere or the time you spent waiting will have been for nothing!!)

(User edit: Specify a folder in the space at the bottom for blender to keep the temporal data. Else it will not work) (Note on user edit: When I specify a name and folder, the water animation disappears.???)

(Yet another User: I tried to Bake my animation, but all it says is something about no images to bake to. I know I need to set a folder or something, but have no idea how!) (You'll get that is you're still in the "Render Options", make sure to go back to the Domain object's "Physics" buttons.)

Noob note:

1. As mentioned above, the baking does not work in the render options panel.

object mode, select domain, physics, fluid, bake

1. Water splash can be observed with 12 frames only.
2. Convert the output of the animation using gimp. Open all files as layers then save as gif.
3. Convert into video using Blender's video sequencer. Refer to Blender 3D: Noob to Pro/Using the sequencer to compile frames into an animation

98.6. Other Fluid Objects

There are two other incredibly useful types of fluid objects--"inflow" and "outflow". The both of these do exactly what they sound like. Inflow objects pour more fluid into the scene, and outflow objects drain it away.

98.7. Keep This In Mind

Blender will make your water its default opaque grey unless you set its color. A good way to make realistic, clear water (as well as glass for that matter) is to edit the color, then apply a simple mirror effect plus transparency effect. After your water has rendered, select the domain in object mode. This selects the liquid in the frame you have just added. From shading [F5] then the 'Materials buttons' (the button that resembles a ball, if when you click on this the colors settings are not up, just press add new under the only panel there), then the panel called "shaders/Mirror transplants/SSS". Go to the Mirror transplants panel, play around with the settings until you get what you fancy. Suitable settings include [the 'Mirror trans' panel of Material:] click 'ray mirror', set rayMir to 0.1 for water, 0.15 for glass, set depth to 7; click 'ray Transp', set IOR to 1.3 for water, 1.5 for glass; then set the Alpha value to 0 while adding an appropriate color from the "Material" panel.

98.8. Notes

This tutorial was just a simple example that leaves much to be desired. Below are some helpful links that will further your understanding of the fluid simulator.

A bunch of Youtube videos demonstrating parameters in the fluid panel (if using Adblock Plus, disable it for this page to work)¹

Some very snazzy fluid examples made by Mpan3 (lots of videos)²

98.9. Site Under Construction

More to come, so check back in a few days!

¹ <http://www.pkblender.it/>

² <http://mikepan.com/motions.htm>

99. Flying Through A Canyon

99.1. Creating the Canyon

99.1.1. Forming from Textures

- Open up a new Blender file and erase the default cube with **X**
- Create a plane (**SHIFT** + **A** -> Mesh -> Plane)
- Press **W** and then specify 100 cuts in the Tool Properties panel (immediately below the Tool Shelf on the left).
 - You could also do this by subdividing the plane twice with the number of cuts both set to 10.
- Scale the canyon up to 5 times its current size
 - Do this by either hitting **S** and typing in 5, or hitting **S** while holding **CTRL** to snap and scale your mesh 5 times.

At this point your mesh should look like something like the picture below.

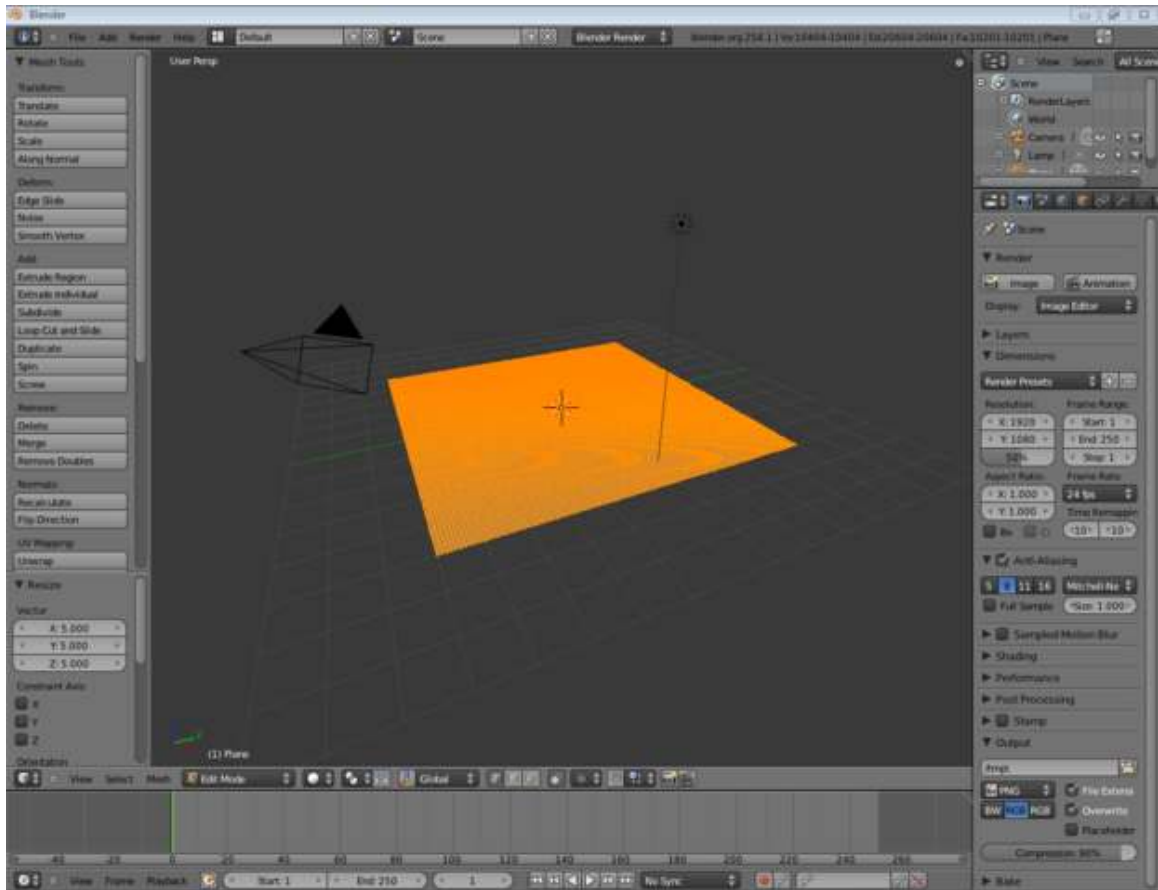


Figure 790

99.1.2. Texturing, Shading, and Deforming Your Mesh

- Go to the Materials tab on the Properties Panel, and click *New*.
- Go to the Texture tab next to the Materials, and click *New*.
 - This texture will be used to create our canyon, so give it an appropriate name. In this tutorial, "Canyon" was used.
- Select the Texture Type to be Wood

User Note: I found that the 'cloud' texture works nicely for creating a canyon, with size 0.45 and depth 2.

- Go back into the Texture tab and uncheck *Canyon*.

We will now displace the vertices based on the texture.

- Click on the Modifiers tab on the Properties panel (spanner icon).
- Click the *Add Modifier* pop-up menu and select "Displace".
- Select your "Canyon" texture for the displacement.

- Go into Object Mode. Notice that the vertices have been displaced! (To see the results better, make sure you're in Solid Mode) However, the material is very jagged. This is because Blender is preserving the original faces on the underlying material, and the original faces don't run in the same direction as the grooves the wood material has created.

User Note: Also works with image textures, i.e. using an image as a texture. Tested using NASA's Blue marble topographic pictures. Go to the end of this tutorial for additional projects.

Your Mesh should now look something like the picture below.

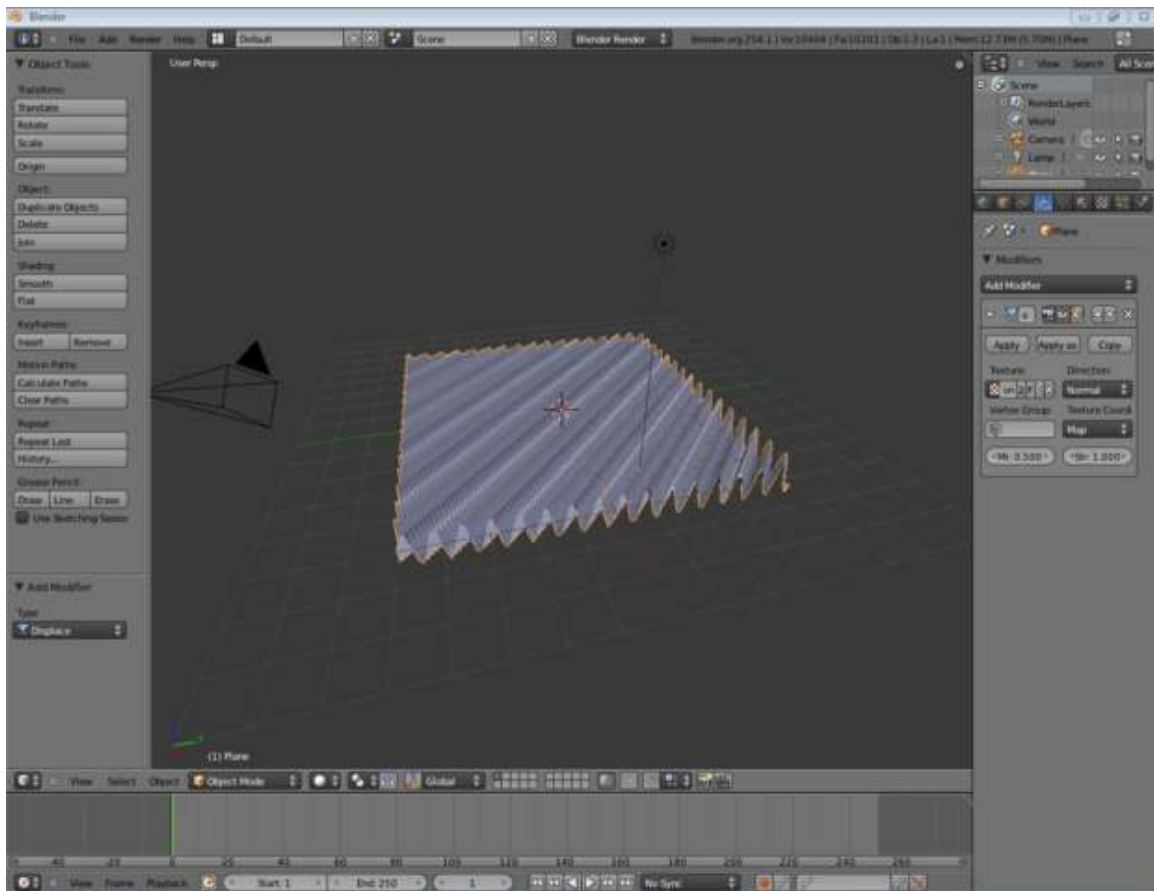


Figure 791

99.1.3. Improving the Look

Now to make the canyon more interesting:

- Go back to the textures tab, under the Wood tab deselect Bands and select RingNoise. Keep the default options for the moment.
- You will now see a "rippling" texture. But it isn't a canyon yet. For this we need to pack the ripples closer together to get more detail per ripple.

- Go into Edit mode (Tab), and make sure all vertices are selected (select one and press L or press A until all are highlighted). Now scale them to 0.2 of the original size. Do this pressing **S** and the minus sign (-) and .2.
- Leave Edit Mode into Object Mode, and scale the Object (as a whole) up to 10.0 of the size; this time by pressing **S** and then 10.
- What just occurred is that you packed the vertices closer to each other, but not the texture. Then, you took the object as a whole (including the Texture) and up-scaled it. This is the difference between Scale in Edit Mode and Scale in Object Mode
- You should now have a semi-boring canyon, but it does have circuits which the camera can float around in.
- To make the Canyon more interesting, play around with the Texture values for Wood. I chose the values Noise Size: 1.0 and Turbulence: 15.00 as they produced a somewhat interesting outer ring.
- You may also want to play around the Deform modifier settings, especially *Strength* which changes the strength of the deformations.
- When you are happy with your mesh, press the *Apply* button under the Deform modifier.
 - The displacements are now permanently baked into the Mesh, and the result should be something like the picture below.

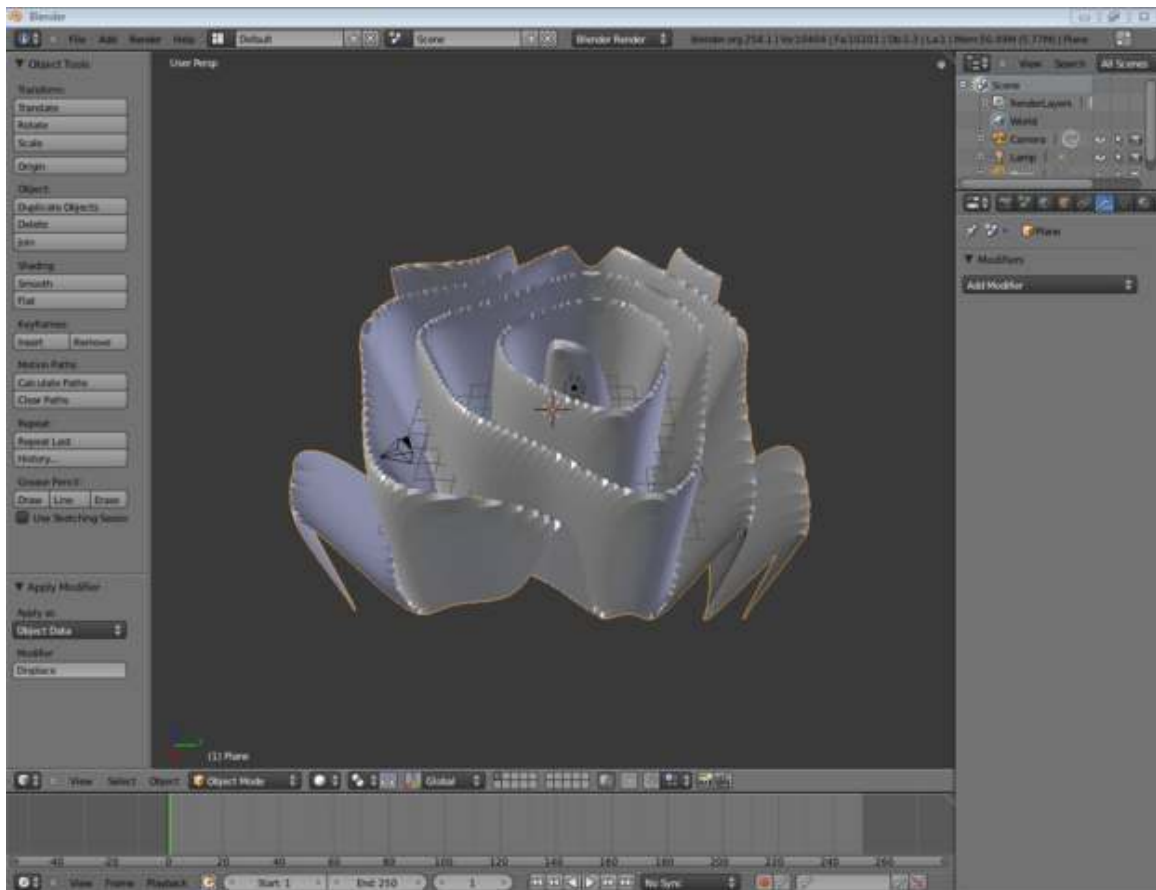


Figure 792

99.1.4. Eliminating the Unneeded

- We will now reduce the number of vertices. Go into Modifiers, and add Decimate. The mesh will appear to change slightly, but it hasn't, except that 4-vertices faces have been triangulated
- Add another Modifier, Sub Surf, and make sure that the Levels is just 1
- Under the Decimate, slowly reduce the ratio and watch the Mesh. Very tiny mesh details will be changed but the Decimate modifier will keep the shape while reducing vertices.
- I got to 0.07 on my mesh before I decided the detail was getting too low. Remember you will be flying through the canyon, not over it, so the detail on the walls of the canyon is more important.
- Hit apply to bake the reduced vertice count into the Mesh. Smaller vertice counts will make rendering faster.
- Increase the number of levels on your Sub Surf modifier and take a good look, you have just created a canyon! You may reduce the levels or remove the Sub Surf entirely afterwards.

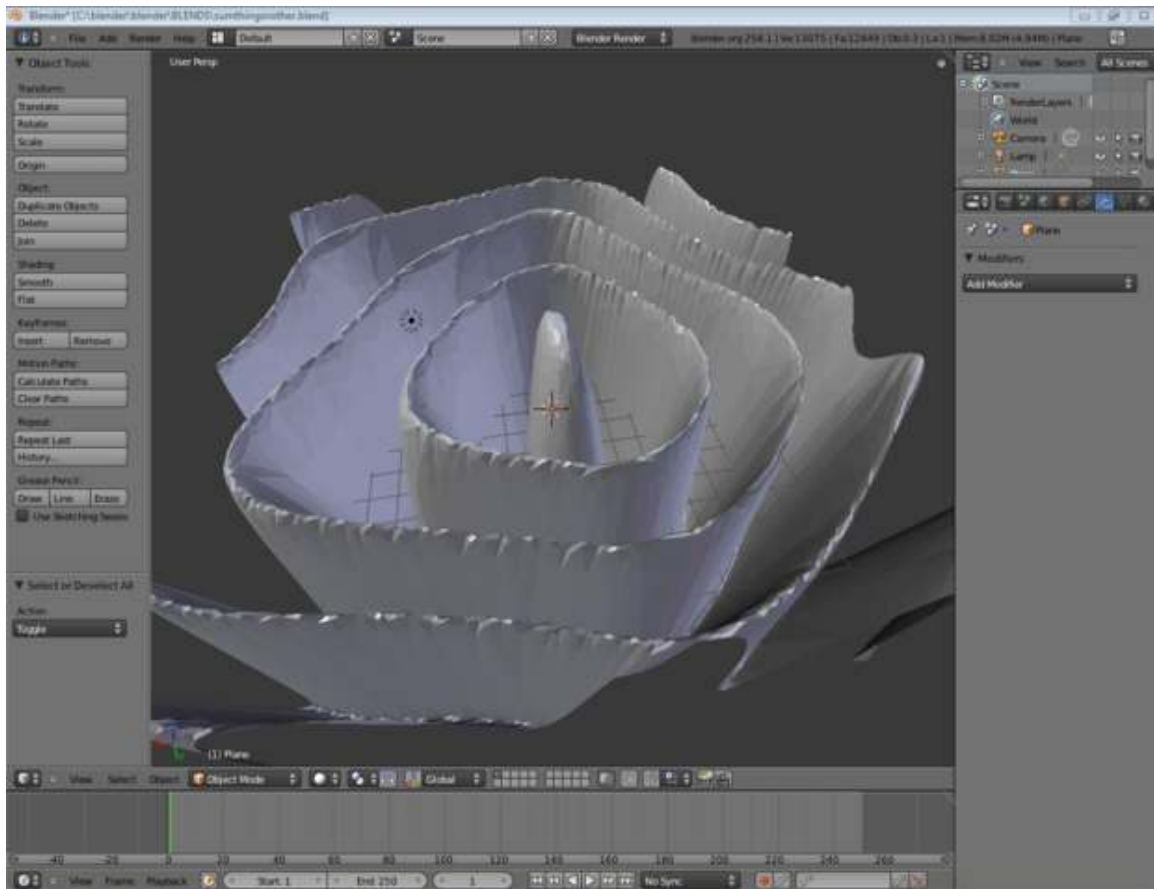


Figure 793

99.2. Coloring the Canyon

99.2.1. A Rocky Material

Method 1: This is one method to create a rocky wall. The result isn't particularly realistic but it is easy to make. Creating materials is a difficult practice, and the first step is to know what you're trying to recreate, perhaps by using actual photographs.

- Create a new material for the cliff
- In the Specular panel under the Materials tab, set the Intensity value to 0.132 (our cliff isn't particularly shiny) and Hardness to 20.
- Under the Textures tab, add a new texture in the first slot.
 - Give this texture the Marble texture type. Under the Marble options, select Hard and set the Noise Depth to 1 and Turbulence to 1.7.
 - Under the Colors tab, activate *Ramp*. The first colorstop should be white with a full alpha of 0.
 - This is meant to give our primitive rock a textured appearance
- In the second texture slot, add another texture.
 - Set this type to Clouds, under the *Mapping* options set the X, Y, and Z sizes to .025, under the cloud options select a noise depth of 2 and activate Hard Noise.
 - This one will recreate some bumpiness and small inconsistencies in the rock
- In the third slot, add another texture
 - Set the type to Clouds
 - In the Colors tab, modify the colors to go from Black, 0 Alpha, to a Brown of your choosing, 0.750 Alpha. Reposition the second color to Pos 0.820
 - This Texture will simulate dirt on the Canyon
- You will definitely want to do a render now. Reposition the camera and lights, and adjust render options to make for a fast render
- Go into Materials, then the textures tab and select the first Clouds texture you entered (should be 3rd on the list if you did not delete the Wood texture used earlier).
- With that selected, under the Map To tab, uncheck the Col, and check the Nor buttons. Set the Nor value to 2.0
 - What we have just done is set the second texture not to affect color, but to affect the angle of the normal in a render. This can be used to reproduce jagged edges on flat surfaces, or waves in water
- Now, select the Marble texture again. Under the Map Input, adjust the sizeX, sizeY, and sizeZ values to 5.0. Do the same for the second texture.
 - You will want to render here, adjust the values as desired. For more layers vertically, increase the sizeZ value
- You should now have an unrealistic, but sort of rocky Canyon wall.

Method 2: This will produce a slightly more realistic texture. Instead of using marble, use clouds. Also, lighten the dark parts of the texture and darken the lightest parts. Make the dark brown parts a slightly lighter shade of brown. This will make your rock wall more realistic.

99.2.2. Lighting the cliff

- Remove all existing lights

- Creating a new light with the type of "Hemi". Center it over our cliffs, and use Scale (s) to scale it down.
- Create another light, with the type of "Sun". Position it where you want (position doesn't matter with Sun), and use Rotate (r) to position its Ray at the Angle you desire.
 - In this example, the Sun was more of an evening Sun. Its light is at a 45 degree angle on one side. This creates some Canyon walls light, some dark.
- Position your camera and adjust the Energy values of your Hemi and your Sun. The Hemi will light up everything, its an easy way to give light to canyon walls that would otherwise be completely black. As a starting point, set your Hemi value at 0.4 and your Sun value at 1.5 and observe how that looks from a few angles. Make adjustments if you desire.

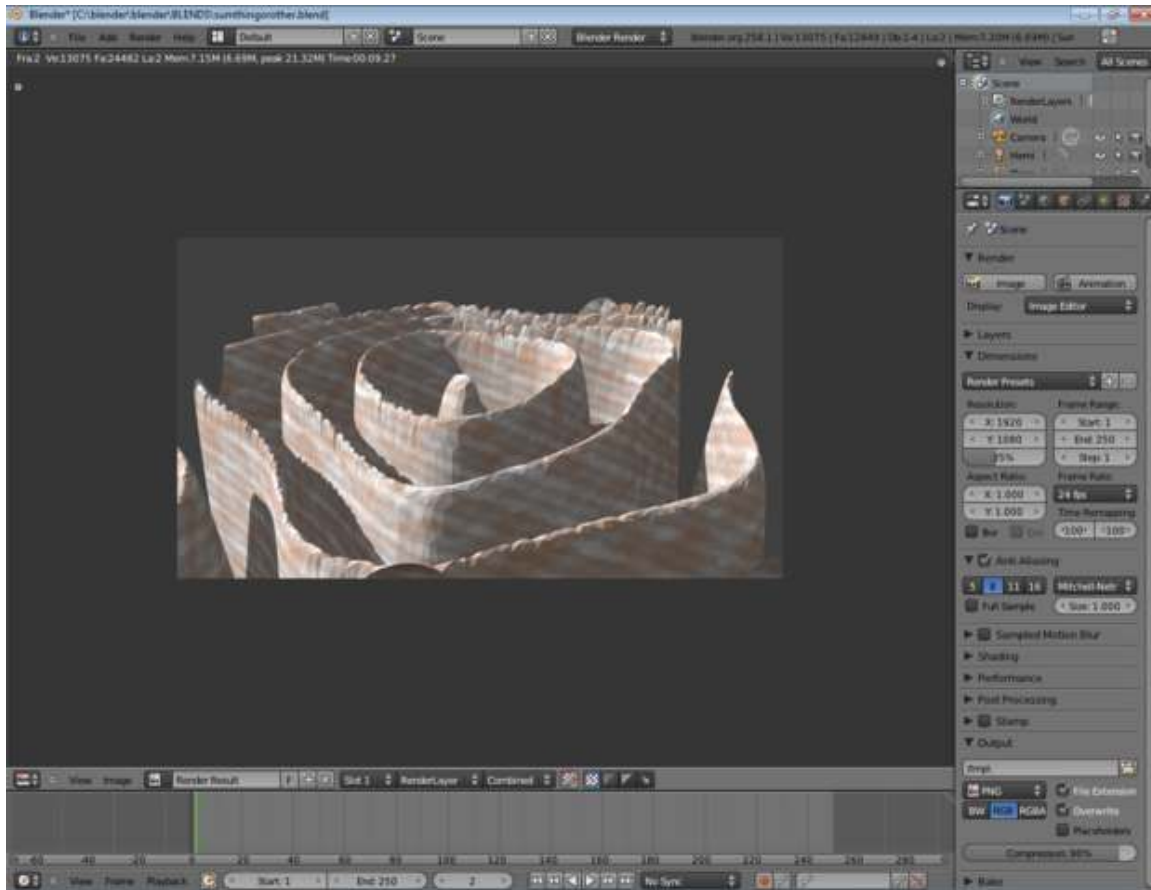


Figure 794

99.3. Additional projects

1. Using NASA's topographic image map at http://earthobservatory.nasa.gov/Features/BlueMarble/BlueMarble_monthlies.php

You have the option of selecting a small portion of your selected topographic map or you can map the entire 360x180 degree panoramic map onto a sphere. Refer to UV Mapping tutorial.

If you just select a small portion of the map of the world(cropping), note the resolution of your cropped image. Create a basic plane and do not scale it yet. Sub divide as many as required to match the resolution of your image. This is in order to get the most out of the topographic image. If you don't bother, this is not essential. The largest subdivision in Blender 2.49b is 120. If you want more than 120 splits, you need to redo the sub divide but the calculation is not straight forward. If you split 2, you get 4 pixels. If you split 3, you get 5 pixels.

Then apply the texture. Choose texture of type "image" and do not disable this texture. NASA's topographic pictures also come in a colourful form but shaded to indicate heights. The reason why you leave the scaling to the last is that the texture is set to map onto the basic plane size of 2x2 units. If you increase this, the textures will be repeated. Adjust the strength level to a suitable level. The default 1 is too large.

100. Using the sequencer to compile frames into an animation

100.1. Using the sequencer to compile frames into an animation

100.2. Preamble

When you render an animation, often it's a good idea to render the individual frames and then compile them into the animation. This has several benefits over rendering directly to a movie file. Some of them are these:

- If you discover a mistake in the render, you can modify sections of the animation and rerender those frames without having to rerender the whole animation.
- If the render crashes because the power went out or, (psst), blender crashed, you can pick up the render at the last rendered frame without having to rerender the whole animation

100.3. Tutorial

- Open your animation in blender

Begin by opening blender to an animation that is ready to be rendered.

- Set the render output to targa format.

press F10 to go the scene buttons and set the output format to Targa. Though you can use any image format you want, I use Targa because it has no compression and is at 100% quality.

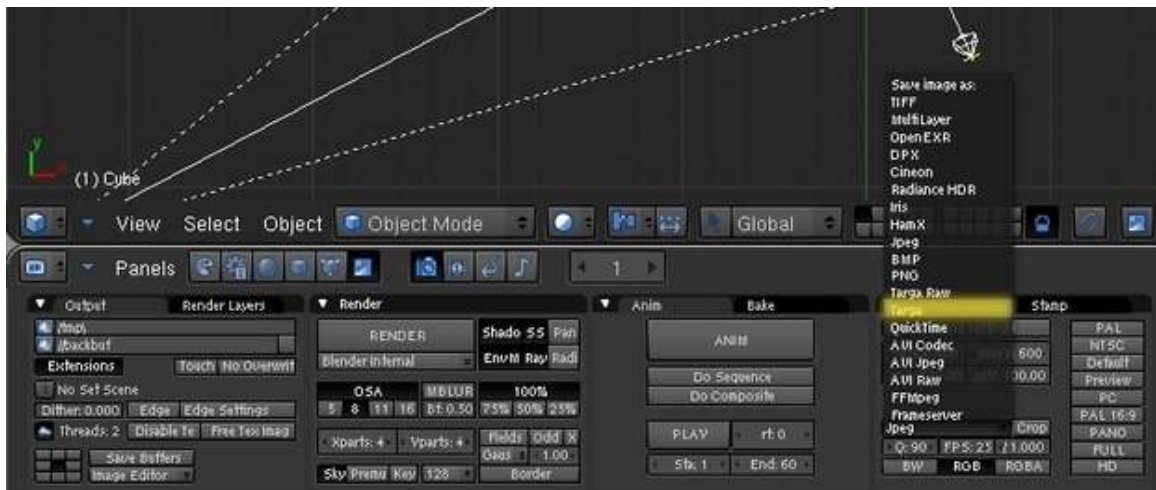


Figure 795

- Set the file output location.

Set the location on your file system where you want all the frames to be saved to.

IMPORTANT: Make sure that there is a slash at the end or else you are setting a filename not a filepath



Figure 796

- Set Endframe to match the animation

If you animated 500 frames, set the end frame to that same number. the default is 250.



Figure 797

- Render the animation!

Simply press the "Anim" button.



Figure 798

100.4. Use the Sequencer to compile the images into a movie file

- Open a new blender and delete everything except the camera.
- Change the 3d window into the sequencer window.

(See image for explanation)

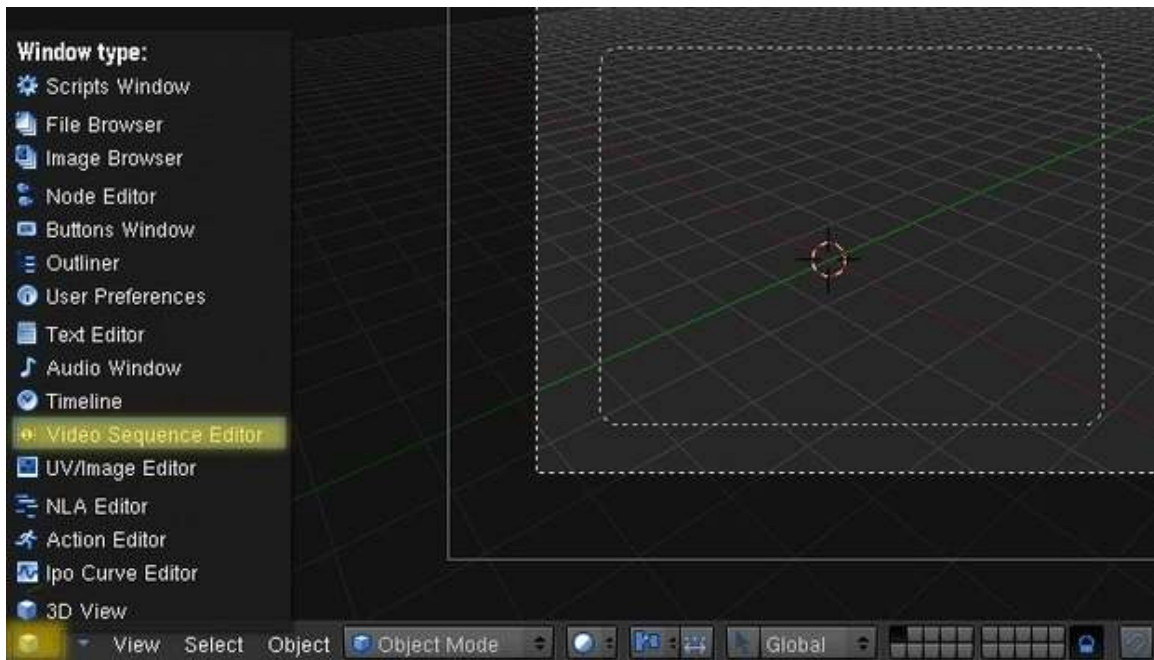


Figure 799

- Add the images to the sequencer.

On the bottom of the sequencer window click Add>>Images then browse to the location where you saved the images, Press Cntrl-A (on linux enter "*.tga" or "*.jpg" in the field for filename) to select all the images and finally press "Select Image" on the top-right. Attached to your mouse cursor will be a long horizontal bar which represents all the images. Move it all the way to the left till the little number underneath it shows "1" and then Left-mouseclick to place it.

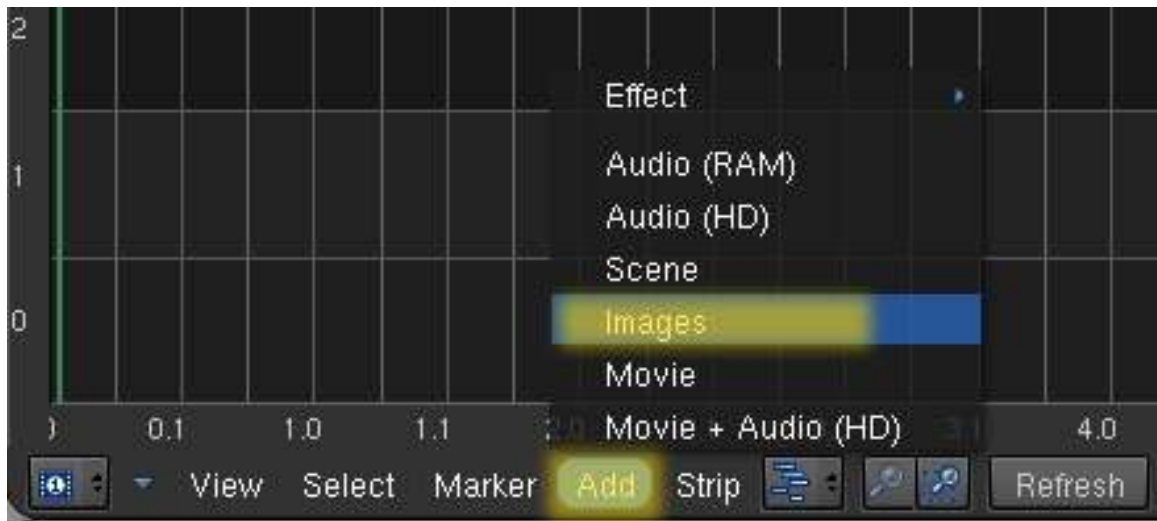


Figure 800

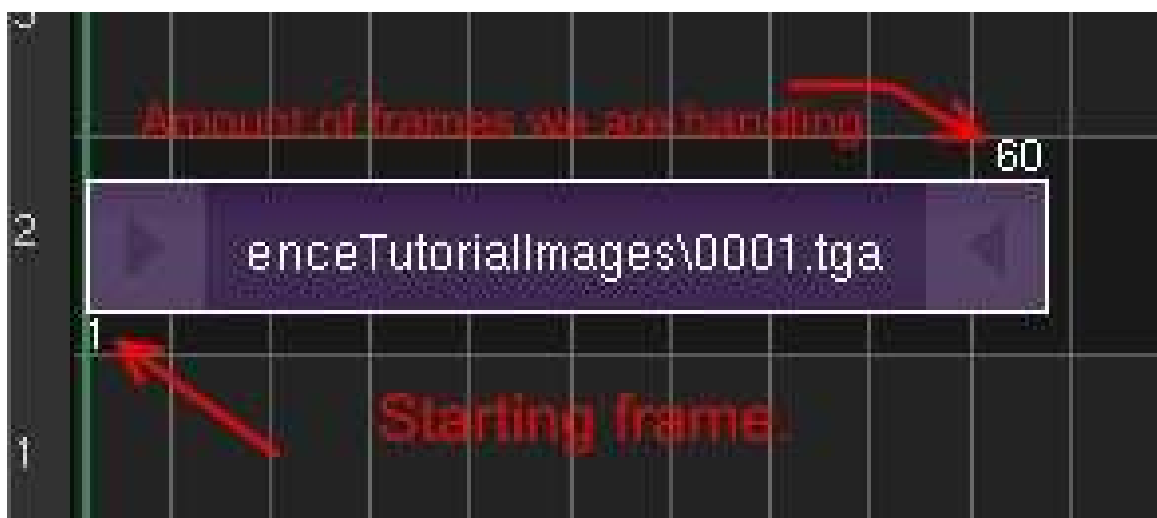


Figure 801

- Set final movie format.

Note that you can also set the location that the movie is saved to by changing the path we set earlier.



Figure 802

- Enable Sequence render and animate!

Depress the Do Sequence button and press the Animate button and wait a few seconds till it's done. Browse to the location you saved your movie and enjoy the animation.



Figure 803

I hope you found this tutorial useful. If there are any errors in the tutorial, *Please* let me know, and I will fix them as soon as possible.

Blend happy and save often, --Radialronnie¹ (talk²) 22:47, 3 November 2008 (UTC)

¹ <http://en.wikibooks.org/wiki/User%3ARadialronnie>

² <http://en.wikibooks.org/wiki/User%20talk%3ARadialronnie>

101. Particle Systems

This tutorial was written for Blender v2.49

Particle systems are used to simulate large amounts of small moving objects, creating phenomena of higher order like fire, dust, clouds, smoke, or fur, grass and other strand based objects. You may also use other objects as a visualization of particles.

Before you start with the tutorials, you should at least take a brief overview about the very extensive manual pages¹ of the particle system. You will find every single parameter explained in the manual if you have the desire to delve deeper ...

Don't forget: particles alone don't do any magic. They are only a placeholder for something nice to view. You have to take care of the visualization also, and that is usually the harder part than to create the particle system.

101.1. The very first particle system

101.1.1. Creating a particle system

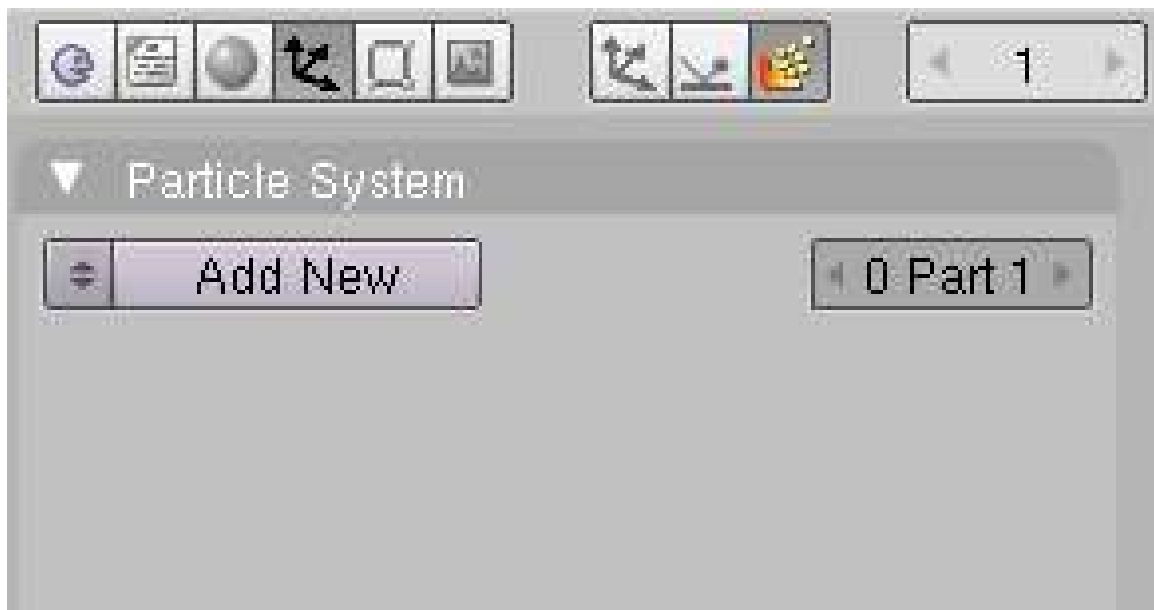


Figure 804 Image 1a: Create a new particle system

¹ <http://wiki.blender.org/index.php/Doc:Manual/Physics/Particles>

To create a particle system:

- insert a plane
- select the object (only mesh objects can emit particles)
- change to the object buttons and to the particle subcontext
- click on *Add new* (**Img. 1a**)

Voila, your first particle system! It doesn't do anything useful now, but we're going to change that on the following pages.

- To see any particles press **Alt-A**. The frames change and you see particles appear. Press **Esc** to stop the animation.

You return to frame 1. The particle movement is cached (stored), so if you play your animation again it will go faster (well, you won't notice any difference in this simple case).

- If you want to stop in a certain frame, press **Space**.
- The shortcut for returning to the first frame is (Shift + Left Arrow)
- To see the particles even better change to wireframe mode (**Z**),

If you change anything in your particle system you always have to return to frame 1, to recalculate the system from start.

Use a *timeline* window to change easily between frames (**Img. 1b**).

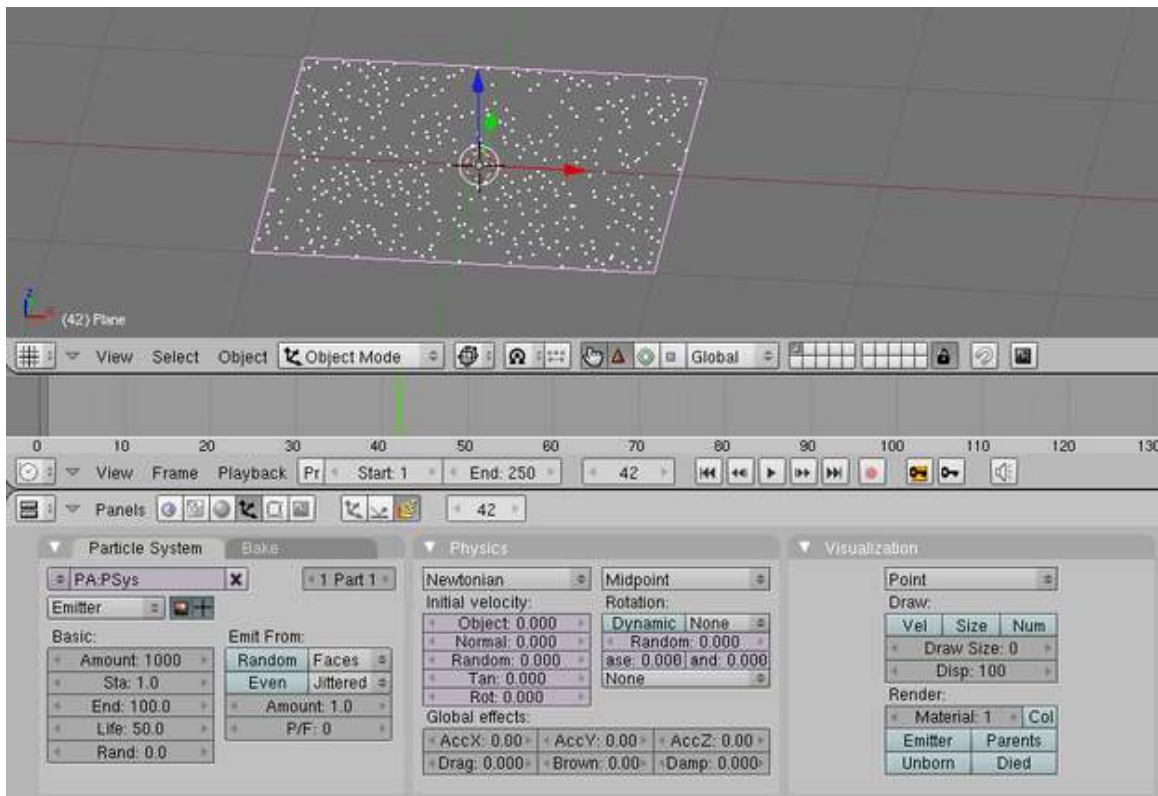


Figure 805 Image 1b: The very first particle system

Note:**Ultra Physics Coolness:**

- Activate *Continue Physics* in the *Timeline* window. Menu: *Playback->Continue Physics*
- Hit the *Play Timeline* button in the *Timeline* window.
Now you can move the objects in the 3D window and the particle system is updated in realtime.
The system is not cached then.

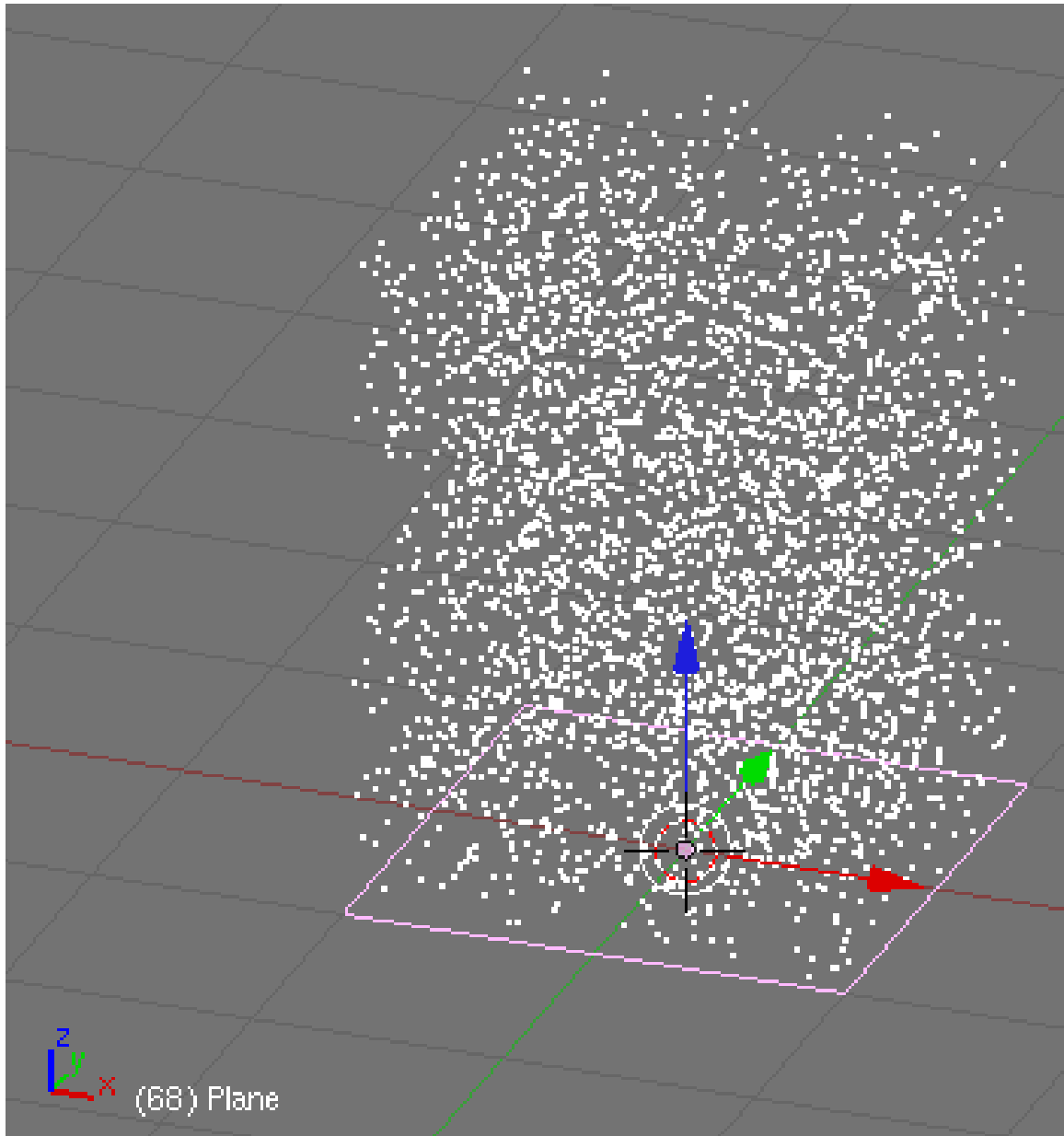
101.1.2. Changing properties of the system

Figure 806 Image 2a: Particles emitted in the direction of the face normal of the plane

The most important properties are probably:

- *Amount*: the total number of particles
- *Sta/End*: the start and end frame of the emission
- *Life*: the lifetime of the particles
- The initial velocity. This is often set with *Normal* causing the particles to get a speed in the direction of the face normal (if emitted from faces).
- A constant global force. *AccX/Y/Z* set's a force in the respective direction.

For now change:

- the *Amount* to 5000
- the *Lifetime* to 100 frames
- the *Normal* value to 1.0.

The Particles are emitted directly upwards (if your plane is orientated that way) and life for 100 frames.

If you render now, you will see a white "Blob". This is the default *Halo* rendering of the particles.

101.1.3. Changing the material of the particles

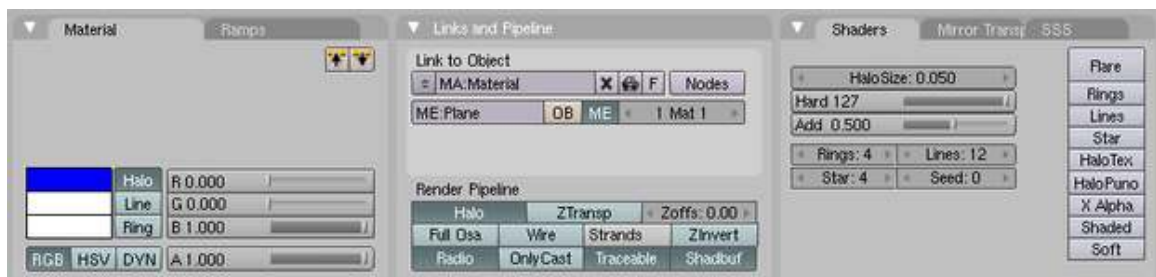


Figure 807 Image 3a: The first simple *Halo* material.

- Change to the *Shading* buttons and assign a material to the plane.
- Activate *Halo* in the *Links and Pipeline* panel.

The *Halo* settings are in the *Shaders* panel. The color of the *Halo* is set in the *Material* panel (see also the Manual on Halos²). Halos are a post rendering effect, that is applied after the scene is finished. So halos can't shed any lights on other objects, they are not rendered behind *RayTransp* materials (like glass).

² <http://wiki.blender.org/index.php/Doc:Manual/Materials/Halos>

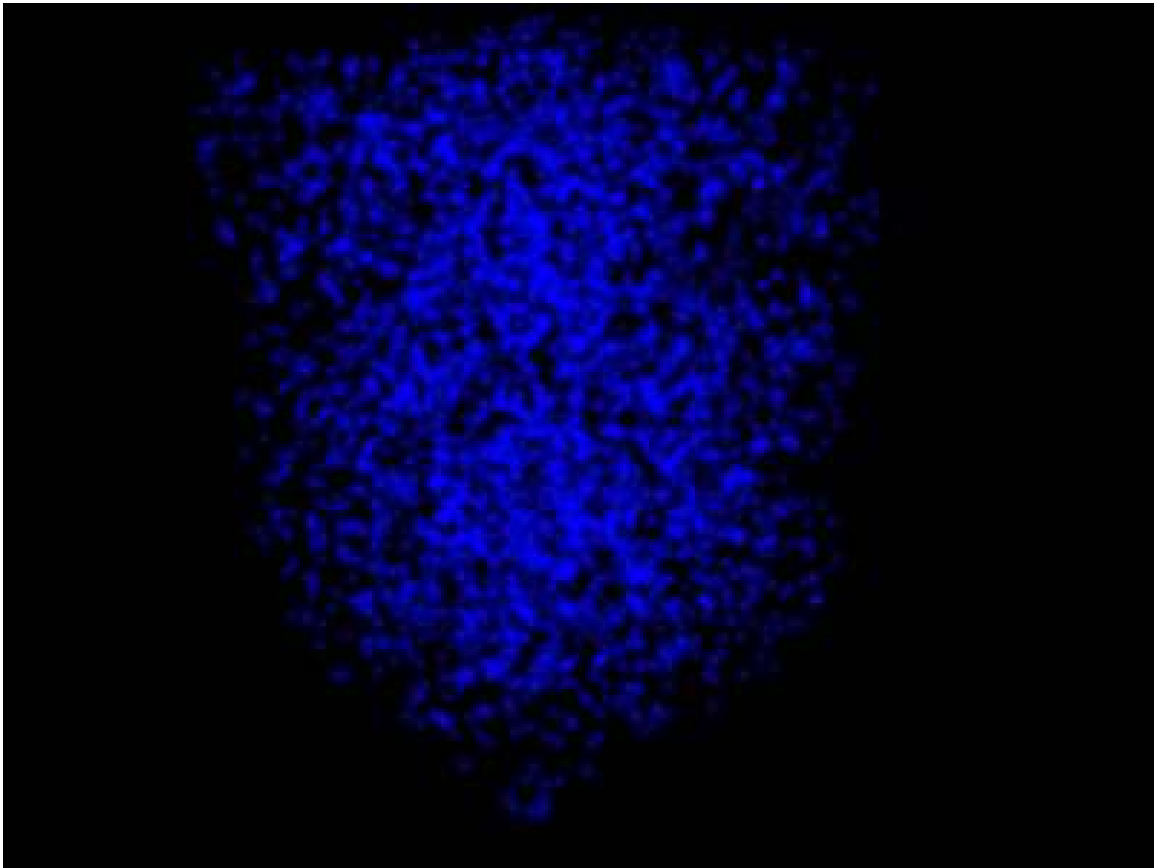


Figure 808 Image 3b: Our first particle system rendered in frame 68

- Set the *Halo* color to deep blue (RGB: 0/0/1)

next go to the "shaders" tab to the right of the "links and pipelines" tab.

- *Halo Size*: 0.05 so each halo is quite small.
- *Hard*: 127 so that each halo has the maximum sharp edge
- *Add*: 0.5 so that the brightness increases where several halos overlap

Set the world color to black and render (**Img. 3b**). Nothing special till now, but that will change soon. So proceed to the next page, where we're going to make some fire.

101.2. Links

- Blender 2.49b Basic Particles Video Tutorial: <http://www.youtube.com/watch?v=dwsR8v5VE4M>

102. Making Fire

This tutorial was made with Blender v2.48a

Figure 809 **Figure 1:** My idea of particle fire.

We're going to create a camp fire with a simple particle system. This tutorial is based on the method described in the [Blender Manual](#)¹. The result of this tutorial is shown in **Fig. 1**, the Blend-File is included at the bottom of this page.

If you need more realistic looking fire, you should use the method described in [BlenderArt Magazine No. 16](#)², though that method is more advanced and uses Compositing Nodes heavily.

The starting point of the tutorial is how fire behaves physically. The flames are made of hot gases. These accelerate upwards due to their lower density in contrast to the cooler air in the environment. Flames are in the middle hot and bright, to the outside they are darker.

1 <http://wiki.blender.org/index.php/Doc:Manual/Physics/Particles>
2 <http://www.blenderart.org/issues/>

102.1. The particle system

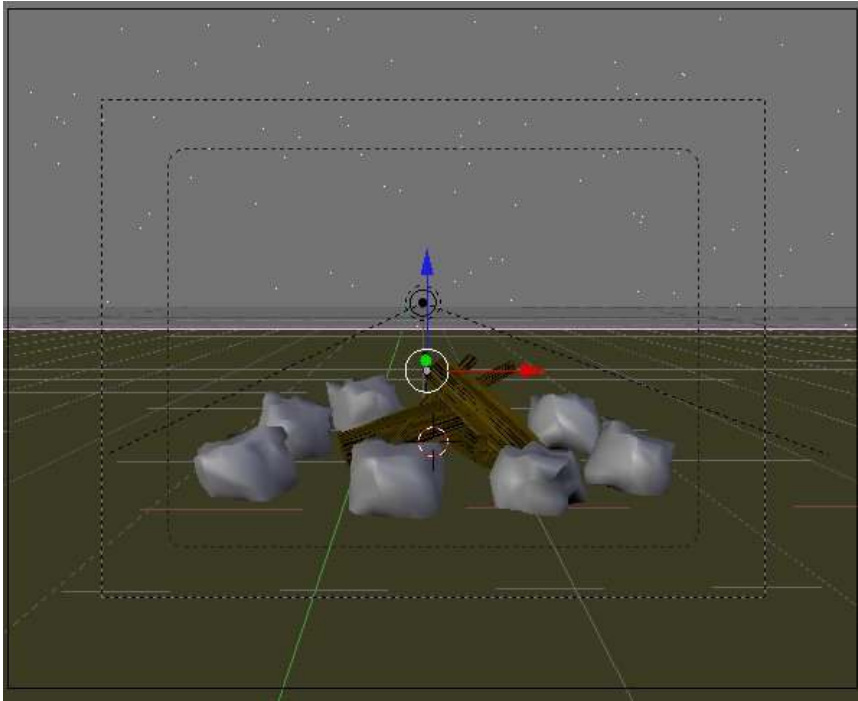


Figure 810 **Figure 2a:** A simple scene

I've created the usual scene with some stones and a few pieces of wood (the wood is by courtesy of Teeth). (**Fig. 2a**).

Noob note: I made the stones by adding an Icosphere of 3 subdivisions, and then used the proportional edit falloff (**OKey**) with the random falloff mode. And then subdivide and smooth.

Noob note 2: For the wood I used a modified Plane with an Image texture created with **GIMP**

- Add a *Plane* in the middle of the stones.

This will become the particle emitter.

- Rename the *Plane* object to "Emitter".

If you use good names you will find it much easier to orientate yourself in your scene later. Having 100 objects named "Cube.something" will make it very difficult to quickly select a desired object.

- Subdivide the plane once in *Edit* mode.
- Change the shape of the plane, so that it's shape equals the base of the fire.
- Change to object mode.

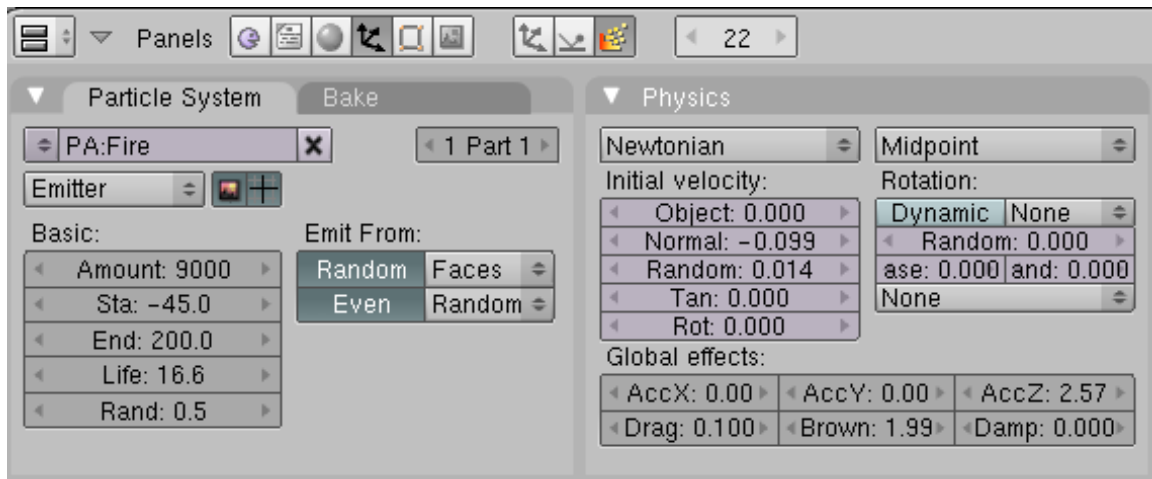


Figure 811 Figure 2b: The particle system

- Change to the *Particle* buttons of the *Object* buttons.
- Create a particle systems (*Add New* in the *Particle System* panel - and make sure the plane is selected!).
- *Type: Emitter* The plane emits the particles.
 - *Amount: 9000* The total amount of particles.
 - *Sta: -45* The simulation shall start before the rendering, to have a fully developed flame in the first frame.
 - *End: 200* The simulation shall last 200 frames - here: the particles are emitted till frame 200.
 - *Life: 16.6* I've adjusted the lifetime of the particles to their speed. Both parameters together regulate the hight of the flame.
 - *Rand: 0.5* The lifetime is changed randomly.
- *Emit from:*
 - *Random*
 - *Faces*
 - *Even*
 - *Random*

This creates particles with a random distribution on the faces of the emitter object.

Note:

Please note, that the particle simulation is only fully calculated if the bake parameters in the *Bake* panel match the positive lifetime of the particles. If you want to run the simulation longer than 250 frames, you have to increase the *End* frame in the *Bake* panel as well. This is independent whether you want to bake or not.

The movement of the particles is controlled with particle physics. You set the *Initial Velocity* and let the physics do the rest.

- *Normal: -0.099* The particles are emitted slightly against the direction of the face normal. This leads to a bit wider fire at the base.
- *Random: 0.014* This creates a random start velocity as well in speed as in direction (you could use a texture to randomize only the speed, see the discussion page for that).

After you have given the particles an initial velocity they are moved by forces.

- *AccZ*: 2.57 A force in positive Z direction (upwards).
- *Drag*: 0.1 Air drag decelerates the particles.
- *Brown*: 1.99 Random movement simulates agitated air movement.



Figure 812 **Figure 2c:** Particles without material

The particle system is finished. Until now it doesn't look like much (see the white Blob in **Fig. 2c**). Therefore the emitter will get a material, this material will be animated.

102.2. Material

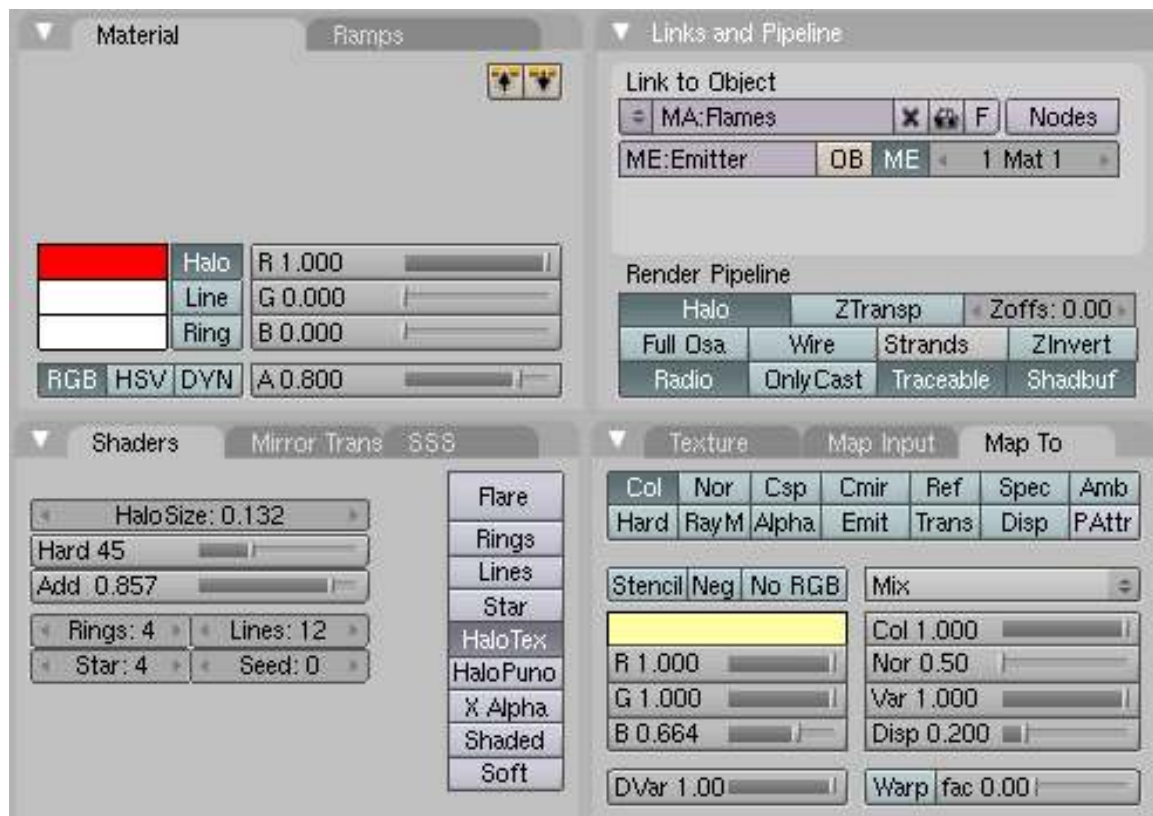


Figure 813 **Figure 3a:** Material for the emitter object.

- Create a new material for the emitter, call the material *Flames*.
- Activate the *Halo* button in the *Links and Pipeline* panel. Else we couldn't set the particle parameters. The particles would be rendered with the default Halo values.
- *Halo* color: 1/0/0 (red)
- *Alpha*: 0.8 The particles shall always be a bit transparent.
- *HaloSize*: 0.132 I wanted many, but fairly small particles.
- *Hard*: 45 The transition from fully transparent to fully opaque.
- *Add*: 0.875 Several *Halos* over each other combine their power. This makes the fire in the center really bright.
- *HaloTex*: A Halo can bear an individual texture, but only the texture in the first texture slot is evaluated.

To give the Halo a bit more structure, give it a texture:

- Add a new texture in the first texture slot.
- *Map To*:
 - *Col*
 - Color: Bright yellow (1/1/0.664)



Figure 814 Figure 3b: Adjusted texture

- Use an adjusted *Clouds* texture with a *NoiseSize* of 0.6.

102.2.1. Animation of the particle material

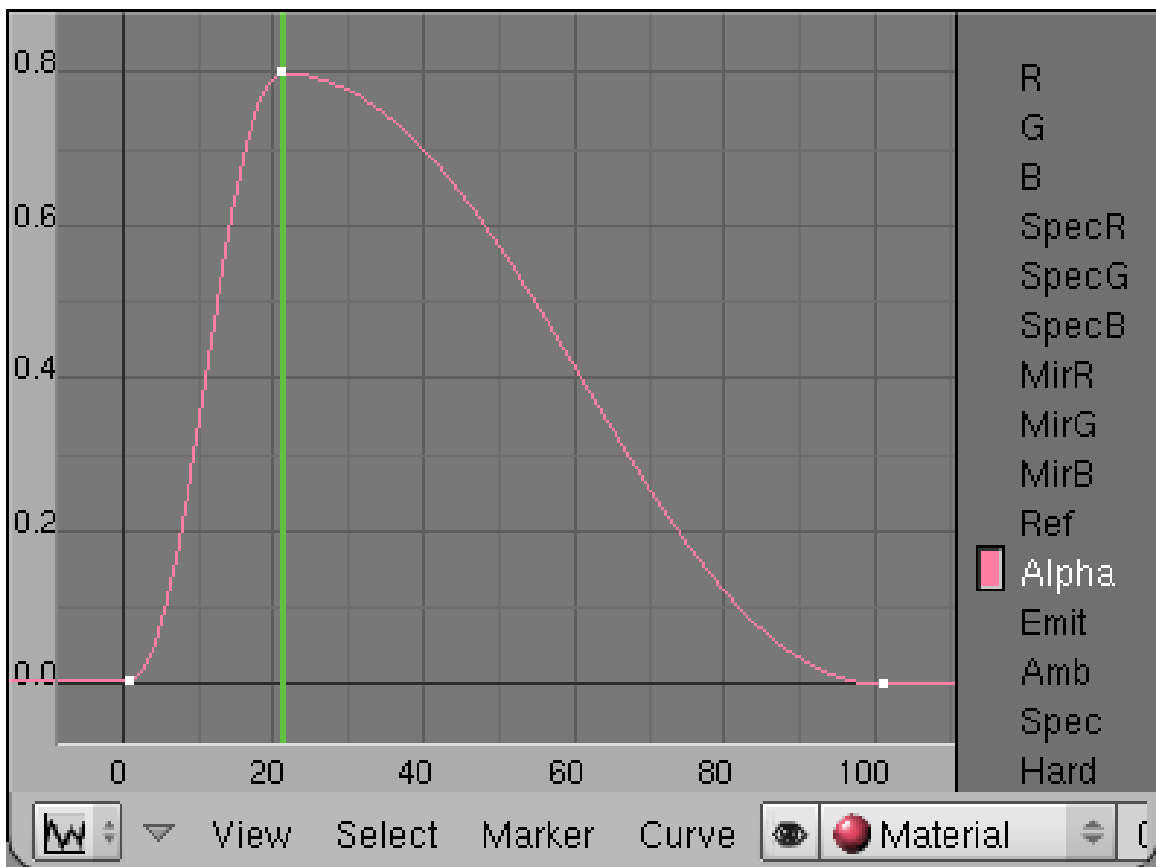


Figure 815 Figure 3c: Animation of the *Alpha* value

The particles "pop" into life and vanish suddenly. We should change that. Therefore we're going to animate the *Alpha* value of the particles.

- Make sure the material buttons are visible in the buttons window.
- Change to frame 21, move the mouse cursor over the button window and press **I**->*Alpha*. This is going to be the maximum visibility of the particles.

- Change to frame 1. Change the *Alpha* value to 0 and insert the next key.
- Change to frame 100. Change the *Alpha* value to 0 again and insert the third key.

If you want to see the IPO curve in the *IPO Editor* window you must change the *IPO Type* selector in the window header from *Object* to *Material*.

Note:

Note: An animation of particle material is mapped from the first 100 frames to the lifetime of the particles. I.e. if the material is faded out during the first 100 frames (the ipo curve is 100 frames long) the particle will be faded out during it's own lifetime, no matter how long that is. This holds true however only for *Point* visualization of particles, not for object visualization.

The *Alpha* value therefore changes during the individual lifetime of each particle from 0 to 0.8 and back to 0 (**Fig. 3c**).

102.3. Rendering

Our particle animation is finished.

- Change the end frame in the *Anim* panel of the rendering Buttons to 200 and click on *Anim*.

Note: If after rendering your particles are too small, such that the fire doesn't look realistic, try increasing the *Halo size* slightly. I used 0.300 instead of 0.132

To actually let the fire glow you have to use one or more lamps and animate them as well. But that would be part of another tutorial ...

103. Furry

This tutorial was created using Blender v2.48a. There is an older version of this page created with Blender v2.40¹.



Figure 816 **Figure 1:** The result of this tutorial: some furry thing

This tutorial deals with fur, i.e. lots of relatively short hairs covering a body. We will use particles to create the fur, and discuss a few aspects here:

- How to determine the length and the thickness of the hair.
- How to determine the place to grow the hair.
- How to color hair.
- How to render efficiently.

The particle system is far too complex to show more than one method in this tutorial. You can achieve many of the same effects shown here in different ways.

¹ <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FFurry%20.40>

103.1. The emitter

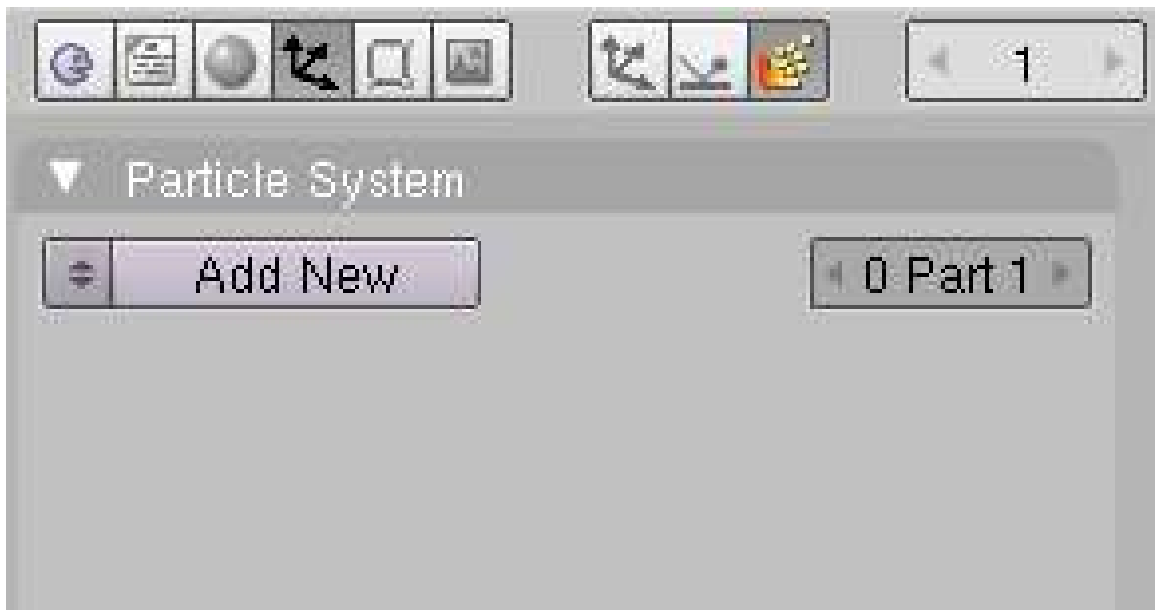


Figure 817 Figure 2a: Adding a particle system.

- Remove the cube.
- Add an *UVSphere*. This will become our emitter.
- Change to the *Particle* buttons in the *Object* buttons (**Fig. 2a**).
- Click on *Add New*.
- Rename the particle system to "Fur".
- Change the particle system type to *Hair*.

A *Hair* particle system has a lot of specialties, the most important thing is that we can edit the particle "motion" by hand if we want to. Apart from that normal particle physics apply, so everything a particle does hair can do also and vice versa. A particle hair shows the way of the particle during it's lifetime at once. To do that efficiently not every single frame is rendered as a point, but a certain number of control points are calculated. Between these control points there will be drawn an interpolated path. The number of control points is the number of segments + 1.

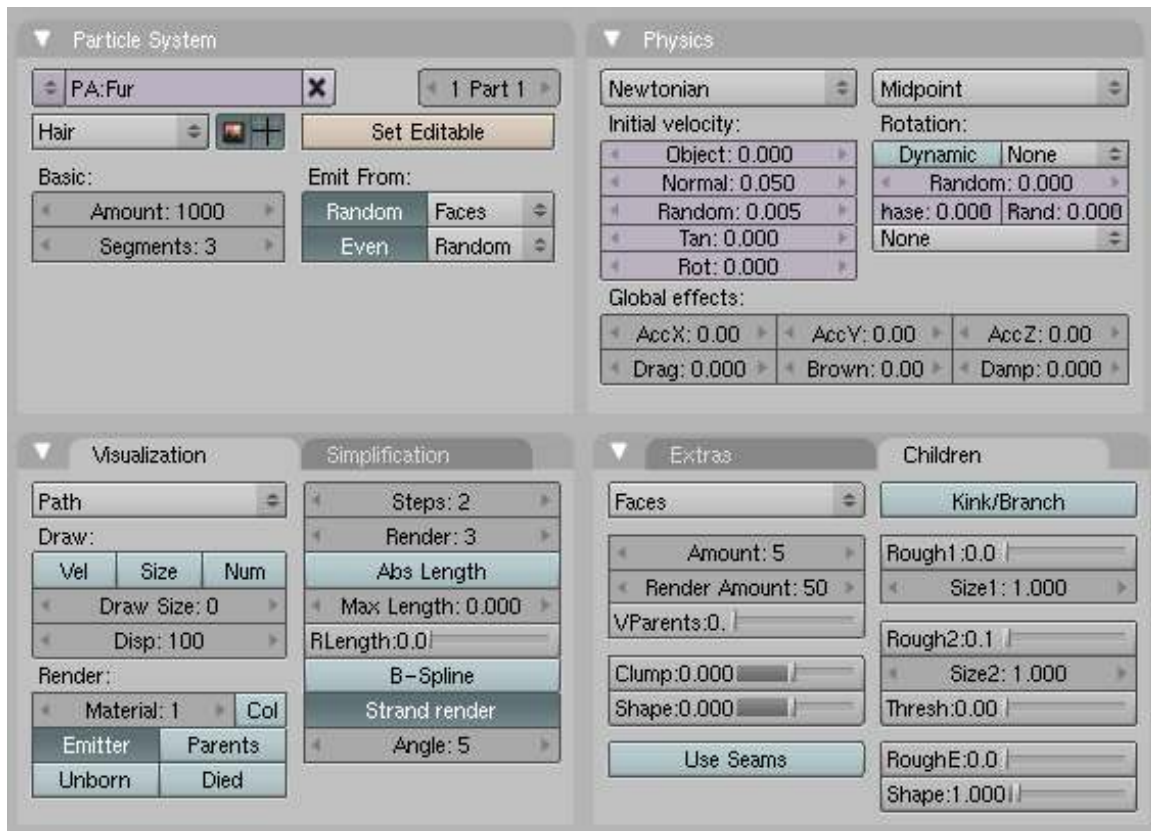


Figure 818 **Figure 2b:** Particle system settings

For fur you need lot's of particles, like 1.000.000 upwards. This will hurt us badly if we have to deal with so many particles in the 3D window and want to render it. Therefore we will create the great amount of particles with so called "children", that mimic the behavior of their parents. The amount of particle parents should be as low as possible, but you need a certain amount to control the distribution of the hair. We will also use as little control points as possible, three segments should be enough for short fur.

- Set the *Amount* to 1000.
- Set the number of *Segments* to three.
- *Emit from:*
 - *Random:Faces*
 - *Even :Random*

This will create a nice, uniform distribution.

Let the hair grow - the hair shows the path of the particle:

- Set *Normal* to 0.05.
- *Random* 0.005

Nothing special here: the hair grows in the direction of the face normals. Length and direction are a bit randomized.

The *Visualization* type changes automatically to type *Path* if you select a hair particle system. If you would render now, you couldn't see the emitter object any more.

- Activate *Emitter* in the *Visualization* panel.
- Activate *Strand Render*.

The *Strand Render* (which I have baptized *keypoint strands* to differentiate from the "normal" *polygon strands*) renders the hair strands extremely efficiently and magnitudes faster than the normal strand. It is the only way to handle many hairs in terms of memory consumption. But it has a few disadvantages:

- They are not seen by raytracing, so you don't get raytracing reflections and no raytracing shadows. You can use environment mapping to compute the reflections and *Spot Lamps* with buffer shadows for the shadow.
 - If the hair is very thick (like 1 BU) sometimes the shape is not correct.
- Activate *Children from Faces*.
 - *Amount: 5* This is the amount of children particles for each parent.
 - *Render Amount: 50* This is the amount of particles during the render.
 - *Rough 2: 0.1* Random variation of the shape of the particles. So the hairs will not stand plain upright and appear a bit curly.

The parent particles are not rendered by default, so now we have 5.000 Particles that render on my old machine in 6 seconds. If we use 1.000 children we have 1.000.000 particles, that need approx. 1 GB of RAM and render in 1:42 minutes. If you render *keypoint strands* with *Children from Faces* you can also use *Child simplification*, which will reduce the amount of particles on objects far away from the camera automatically.

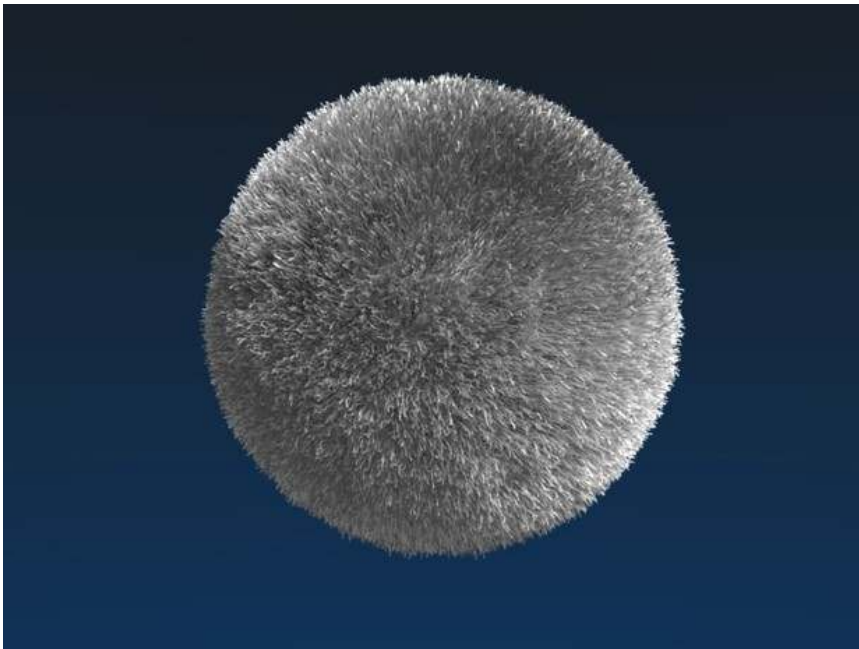


Figure 819 Figure 2c: The first render without material.

Now we should change the lighting to get a preview.

- Select the lamp.
- Change to the *Lamp* buttons.
- Change the lamp type to *Spot*.
- Change the shadow type to *Buf. Shadow*
- Change the buffered shadow type to *Classic-Halfway*.

This is a great shadow type that renders keypoint strands very well and creates fewer artifacts than *Classical* (In my opinion). I have inserted two other lamps and used a classical three point lighting for the first rendering (**Fig. 2c**).

103.2. Material

In the material buttons you can set different aspects for the strands:

- their width and form
 - the used shader
 - the base color
 - a texture along the strand
 - different particle attributes like length, density or roughness
- Add a material to the emitter.
 - Name the material "Fur".

103.2.1. Strands Shader

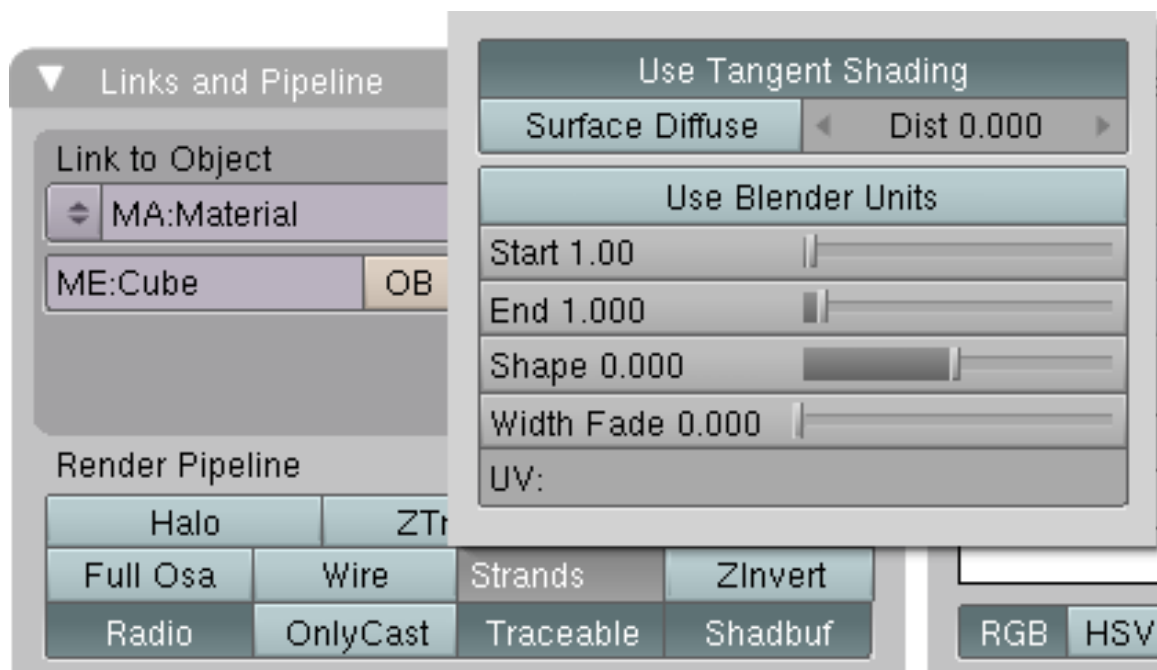


Figure 820 **Figure 3a:** *Strands* settings in the material buttons.

The default strands settings for *Keypoint strands* are shown in **Fig. 3a**. Take a look in the Manual about Strands² for an explanation of all settings.

- Change the *End* value to 0.25, this will make the hair more spiky (not shown in **Fig. 3a**).

Note:

Surface Diffuse: It might be easier to achieve a good lighting environment if you check *Surface Diffuse*. With *Surface Diffuse* turned on, hair does not show as many highlights, which may be too bright without it. I've seen many great examples of fur that use this parameter. However, you have to decide right at the beginning whether you want to use it because it affects all the other material settings. This is something you have to decide for yourself.

103.2.2. Giving the hair its base color



Figure 821 **Figure 3b**: Fur color texture

Strands are rendered with the material of the underlying face/vertex, including shading with an UV-Texture. Since you can assign more than one material to each face, each particle system may have its own material and the material of the underlying face can be different from the material of the strands. We will use an UV texture and use it for the surface of the emitter as well as for the color of the hair.

- Change to *Front* view in the 3D window (*View->Front*).
- Make sure you are in *Orthographic* view mode (also in the *View* menu).
- Change to *Edit* mode of the sphere.
- Press **U** to unwrap, select *Sphere from View*. This is a quick and well working method to correctly unwrap a sphere the easy way.

You don't need to assign a texture in the *UV/Image Editor*, we only need the coordinates now.

² <http://wiki.blender.org/index.php/Doc:Manual/Materials/Properties/Strands>



Figure 822 **Figure 3c:** Emitter with color texture.

- Add a texture to the material, name it "FurColor".
- Set *Map Input* to *UV*.
- Go to the *Texture Buttons* and set *Texture-Type: Image*.
- Load an image texture. I have used the image in **Fig. 3b**.

Normally I would just stop here, I think the material is good enough. But if you want to make the fur more fluffy and soft, you should add a second texture along the strand, which changes the alpha value.

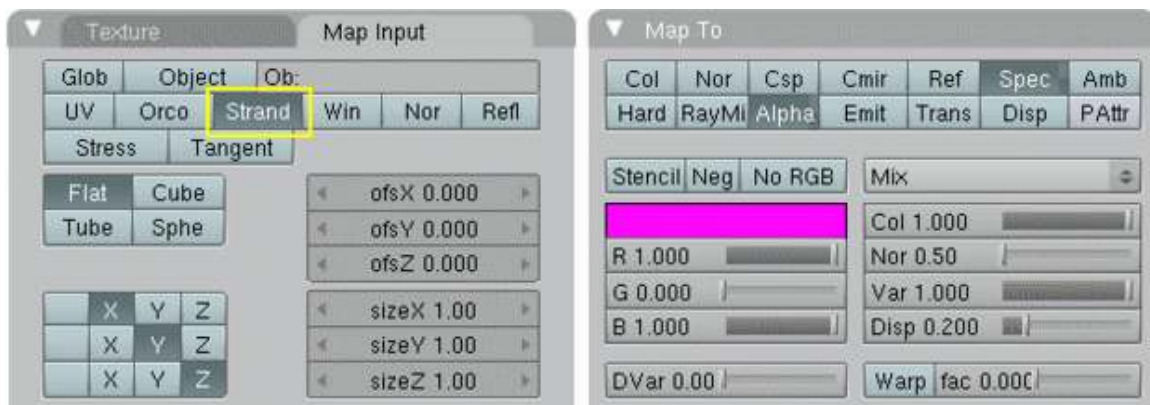


Figure 823 **Figure 3d:** Settings for a texture along the strand.

If you want to do that:

- Activate *ZTransp*.
- Add a second texture.
- *Map Input: Strand*
- *Map To: Alpha and Spec, DVar=0*

- Use a blend texture (*Linear* or *Quad*)

You can change all other properties this way, for example the color along the strand (bleached tips).

103.3. Changing Hair length with a texture

At first I will show you how to render the emitter mesh with a different material than the strands. Then I will show how to change the length of the hair with a texture semi-interactively.

- Change to the *Editing* Buttons.
- Change to *Edit* mode.
- In the *Links and Materials* panel click on *New* in the material section (see **Figure 4a**).

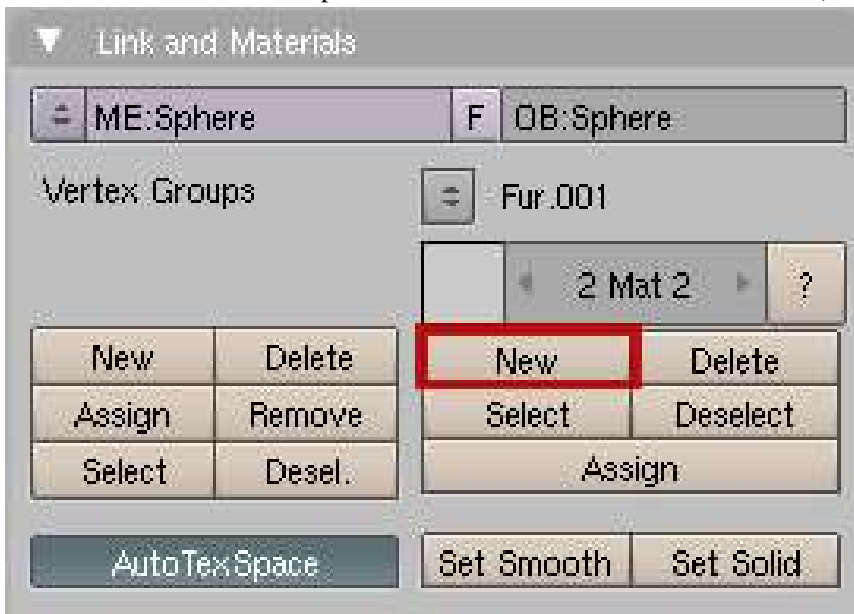


Figure 824 Figure 4a: Button to click

- Make sure all vertices are selected.
- Click on *Assign*.
- Change back to *Object* mode.

Now the emitter bears a second material.

- Return to the material buttons.
- In the *Links and Pipeline* panel click on the **X** next to the material name (*Deletes link to this Datablock*).
- Add a new material.
- Name it *Emitter*.

Now you have a new material on your emitter object. Since the particle system uses material no. 1 you can use different settings for the emitter.

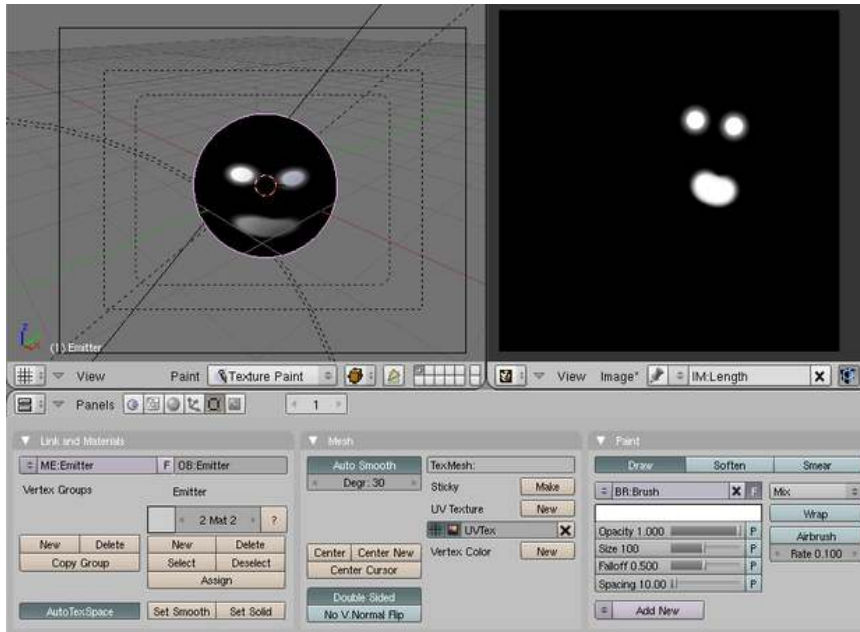


Figure 825 Figure 4b: Texture painting

We have already unwrapped the emitter, this is something that will probably be the case also for any real models. Now we will use an UV-Image and texture painting to determine the hair length.

- Split the 3D window.
- Change the right hand side to an *UV/Image-Editor* window.
- Change to edit mode of the emitter object.
- In the *UV/Image-Editor* use *Image->New...* and confirm the default settings. This will create a new image, that we will paint on.
- Click on the package icon in the windows header of the *UV/Image-Editor* window. Confirm.
- Change the object to *Texture Paint* mode. **Noob Note:** *In the 3D Window, click on the drop down menu. Change it to **Texture Paint**.*

Now you see the texture on the object.

- Paint a structure on the object.
- Change back to object mode.
- Go to the *Material* buttons.
- Change the active material to *Fur* (click on the arrows in the *Links and Pipeline* buttons where it reads *2 Mat 2*).



Figure 826 Figure 4c: Material settings for setting the particle length with a texture.

- Create another texture. Name it *FurLength*.
- Set *Map Input* to *UV*.
- Set *Map To*
 - Turn *Col* off.
 - *PAttr*
 - *Length*
 - *DVar=0* All the white areas on the texture will produce a particle length of 0.
- Load the image texture that we have painted.



Figure 827 Figure 4d: Controlling particles with a texture: result

The result is shown in **Fig. 4d**, you can also see the particles change in the 3D window.

There would have been other ways to achieve this result, e.g. with vertex groups or with particle editing. But I like to work with textures, because you have very fine control and may change the strength of the effect at any time. Vertex groups don't allow for such fine control or you need very many vertices in the emitter. Particle editing (what we will do in the next step) is lost if you change the base particle settings late on, and you can't change it's effect so easily.

103.4. Comb it!

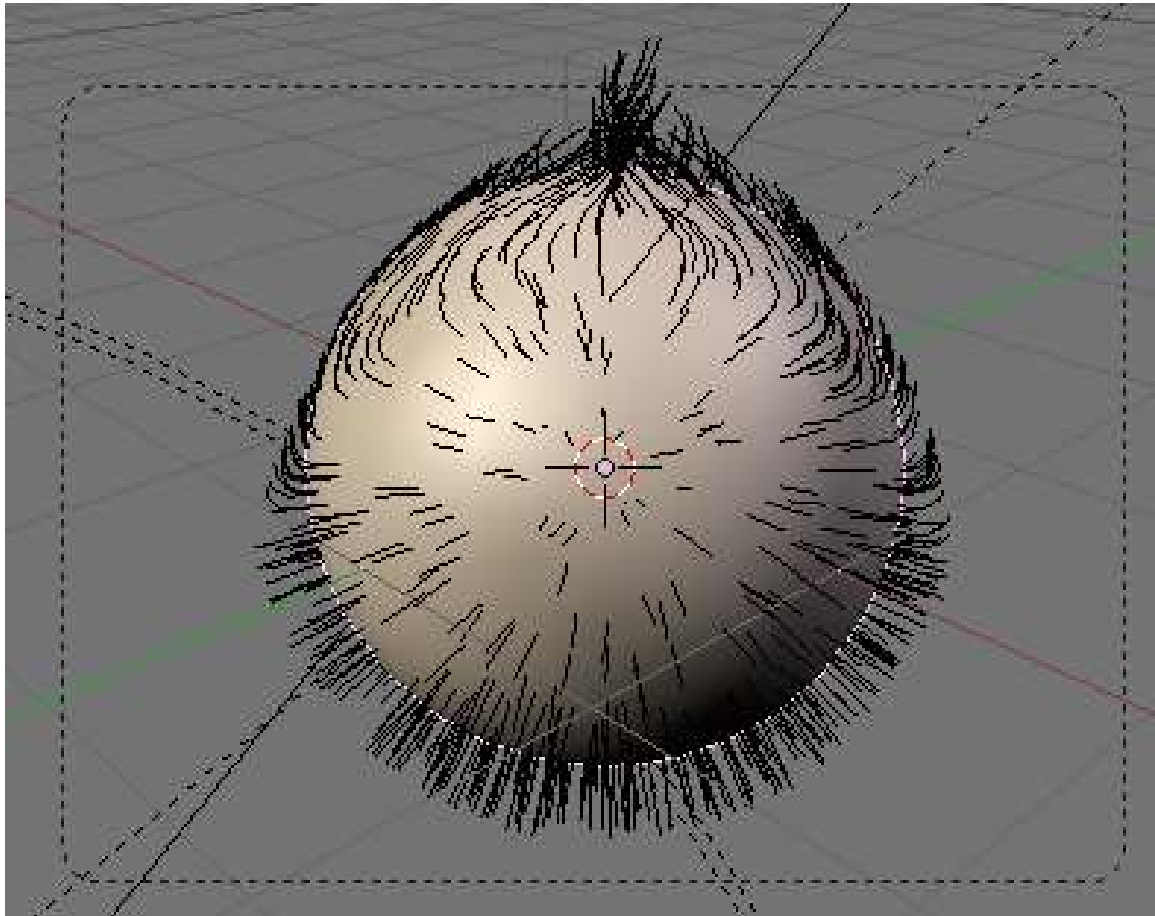


Figure 828 **Figure 5a:** Combing in *Particle Mode*

An effect that is often underestimated is the importance to comb fur in the natural directions. Fur doesn't simply stand upright, and it also doesn't follow gravity (or only to a small amount). So back to the particle system!

- Change back to the *Particle buttons*.
- Click on *Set Editable* in the *Particle System* panel.
- Change to *Particle Mode*.

Particle Mode only appears if you have made the particle system editable, and only hair systems can be made editable. There are a few lifesavers to know when working in *Particle Mode*.

- you can edit only the control points (remember the setting *Segments* from the beginning of this tutorial)

- you can only edit parent particles, so you need enough parents for good control

- Activate *Limit selection to visible*
- Activate *Point select mode* for even finer control.

Both settings are in the window header of the 3D window.

- Open the *Particle Edit Properties* panel with **N** key in the 3D window.
- Select *Comb*.

You have quite a few different tools at hand in *Particle Mode*, see the manual on Particle Mode³.

- Comb the hair following the natural flow.

On the example sphere I have used here it is a bit difficult to tell what the natural flow should be ;-). So I have just very carefully combed and changed the length at a few places a bit. You find the rendered result in **Fig. 1**, the Blend file is linked below.

103.5. Links

- MISSING: The result of this tutorial: http://www.snapdrive.net/files/180816/Blender3D_FurTutorial.zip⁴
- A tutorial in German that shows how to create Grass⁵ with a very similar method.
- Awesome fur shown here: Tiger⁶

3 http://wiki.blender.org/index.php/Doc:Manual/Physics/Particle_Mode

4 http://www.snapdrive.net/files/180816/Blender3D_FurTutorial.zip

5 <http://en.wikibooks.org/wiki/%3Ade%3ABlender%20Dokumentation%2F%20Tutorials%2F%20Partikel%2F%20Gras>

6 <http://blenderartists.org/forum/showthread.php?t=124839>

104. Fireworks

This tutorial was made with Blender v2.48a

Figure 829 **Figure 1:** Some particle fireworks

How to create a firework from particles? We will use cascaded particle systems especially of the type *Reactor*.

Abstract:

- We create a emitter object in the appropriate size.
- Than we use three successive particle systems:
 - the first of the type *Emitter*
 - the second of the type *Reactor*. This system reacts to the death of the emitter particles.
 - the third again of the type *Reactor*. This systems reacts to the proximity of the second systems and thus creates a "drag".
- We create three different materials (one of them animated) and assign them to the three particle systems.

104.1. The Emitter

We use a *Plane* as emitter object. Scale it to your liking. I have used a relatively large particle system, so my plane has a length of 15 BU (Blender units).

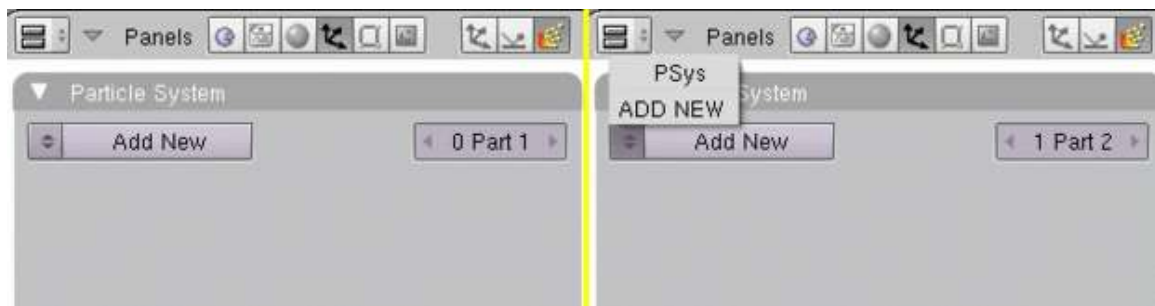


Figure 830 **Figure 2a:** Adding particle systems. First particle system on the left hand side, a second particle system on the right hand side

- Add a particle system (**Fig 2a**, left hand side).
 - *Amount:* 25
 - *Sta:* 1
 - *End:* 250

- *Emit from:*
 - *Random*
 - *Faces*
 - *Even*
 - *Random*
- *Initial Velocity*
 - *Normal: 22*
 - *Random: 7*
- *AccZ: -9.8*
- *Visualisation: Line*
 - *Back: 1.1*

A few particles (25) are created in the first 250 frames of the animation and emitted upwards. Gravitation, velocity and lifetime are adjusted so, that the particles reach the end of their lifetime at the topmost point of their trajectory.

The *Line* visualization lets the particles appear as long drawn-out line.

- *Bake* the particle system. Use 500 as the *End* frame for bake. 300 would be sufficient here, but 500 do no harm.

Noob Note: Bake is in the Bake tab next to particle system tab in particle buttons window.

The plane will get a *Halo* material.

- Activate *Halo* in the *Links and Pipeline* panel.
- *Halo*: color red
- *Halo Size*: 0.421
- *Hard*: 35

I've adjusted the size and hardness of the halo so long until I liked it, so there is no specific reason to use these values.

104.2. Reactor 1

- Click on the arrow next to *1 Part 1* in the *Particle System* panel.
- The settings for the first particle system are no longer shown, now it reads *1 Part 2* (**Fig. 2a**, right hand side).
- Click on *Add New*, now it reads *2 Part 2*.

You just have created a second particle system. Rename this system, a good naming convention will help you a lot to keep the overview.

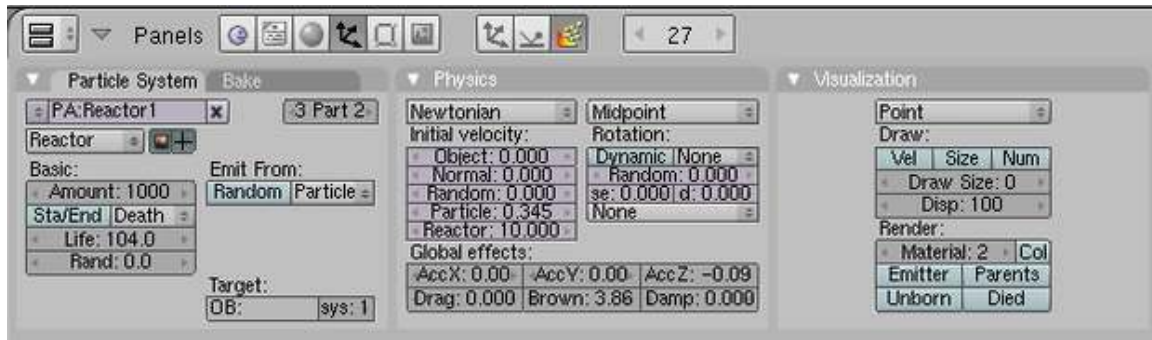


Figure 831 Figure 3a: The first reactor system

- Activate *React on: Death*.
- *Life: 104*
- *Emit from: Particles*

You don't have to set anything for the *Target*. It is not necessary to set the target object if it is the same object as the reactor. You just have to set the number of the target particle system eventually.

The *Reactor* particles react on the *Death* of the particles of the target system. They will be emitted from the point that the particles occupy at their death.

To control the movement of the system, I have made following setting in the *Physics* panel:

- *Particle: 0.345*
- *Reactor: 10*
- *AccZ: -0.09*
- *Brown: 3.86*

Because of the *Reactor* setting the star is moving away from the emitting particles. The *Brown* movement lets the trajectory appear trembling like affected by wind.

- Set the material number to "2" in the *Visualization* panel.
- Bake the system, again use 500 as end frame.

104.2.1. Material 2

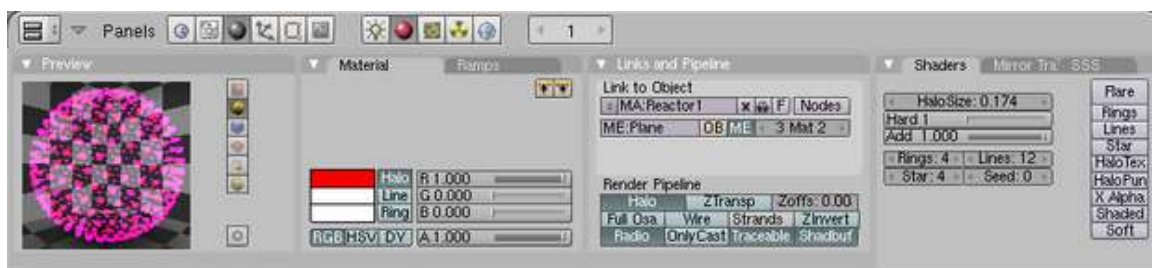


Figure 832 Figure 3b: Material settings for the 1. *Reactor* system

- At first you have to assign a new material in the *Link and Materials* panel of the *Editing Buttons*.
- Use the material settings from **Fig. 3b**,

- The material also gets a *Clouds* texture with *Noise* depth "3", simply leave all other settings unchanged.

The texture is actually superfluous here, but the next particle system is given the same texture, and colors should be adjusted a bit.

- Animate the *Alpha* value of the material.
 - Set the first Ipo key in frame 1 (*Alpha=1*). To do that move your mouse cursor over the *Buttons* window. Press **I**->*Alpha* (and change ipo type to material in the ipo curve editor if you are using that).
 - Set the second Ipo key in frame 86 (*Alpha=0.75*).
 - Set the third key in Frame frame 101 (*Alpha=0.0*).

The second particle system is faded off relatively quickly, but the particles don't disappear suddenly.

104.3. Smoke Trail

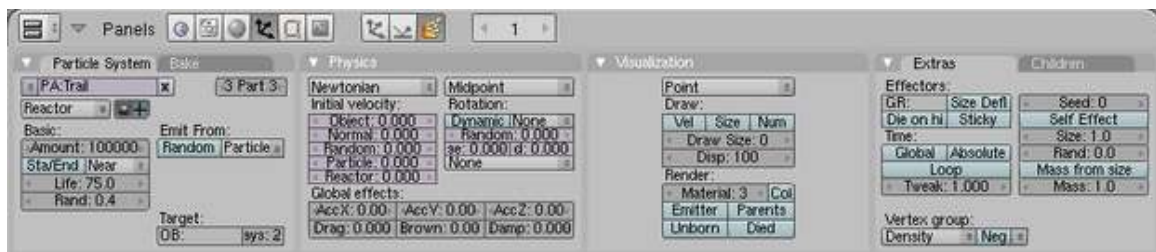


Figure 833 Figure 4a: Particle system for the trail

The third (and last) particle system is again a *Reactor* system and is reacting to the second system. It's going to react to the *Nearness* of the particles. Without moving the variation of the particles is created with a texture and a random variation of their lifespan. What Blender considers as close can be changed with the particle *Size*.

- Create a third particle system (like you created the second).
- Use the settings from **Fig. 4a**. Important are the settings:
 - *Amount:* 100000
 - *React on:* *Near*
 - *Emit from:* *Particles*
 - *Life:* 75
 - *Rand:* 0.4
 - *Target Sys:* 2
 - *Material:* 3
- Bake the system, again use 500 as end frame.

104.3.1. Material 3

Figure 4b: Material for the *Trail* particle system¹

- At first you have to assign a new material (the third) to the object in the *Link and Materials* panel of the object buttons.
- Assign the existing *Clouds* texture from material 2 also to material 3, but select a dark blue as target color (in the *Map To* panel).

104.4. Render

- Render your animation, use end frame 500.

That's all. I hope you were able to follow the tutorial. Criticism and suggestions, write best on the talk page or improve the article directly.

¹ <http://en.wikibooks.org/wiki/File:Blender3D%20Fireworks-Reactor2-Material-2.48a.png>

105. Particles forming Shapes

This tutorial was created with Blender v2.48a

Figure 834 Image 1a: Particles can move to the position of other particles with *Keyed* physics.

Figure 835 Image 1b: Particles can react to force fields of other particles.

I will show on this page two ways to let particles take the shape of other objects.

1. keyed particle physics (**Img. 1a**)
2. a harmonic force field with a damping of 1. The force field is emitted by another particle system, so that each particle is attracted by another particle (**Img. 1b**).

v2.48

The second method is easier to control, but I've found that it may not work exact enough. Even if you use RK4 as calculation method, there are some glitches in the precision of the calculation, so that your particles may not come to a standstill (depending on the size of the particles and their number this may be visible or not).

v2.49

That the particles didn't come to a standstill has been a bug that Jahka has fixed in 2.49. So if you use Version 2.49 and upwards you will find it easier to work with the harmonic force field.

Note:

Plan Ahead!

Make it easy for yourself. If you know beforehand what will happen at each important frame, you will save a lot of time fiddling with the parameters.

105.1. Keyed Physics

So here we're going to use keyed particle physics and quite a few different particle systems.

Basically this is the way to go:

- Use mesh objects and let them emit particles.
- Each system moves to another system because of the keyed physics.
- Choose a good visualization of the particles and animate that also.
- At last the hard part: sync the different animations.

The animation shall last 300 frames, this will be 12 seconds.

- In the first 2 seconds the word "Blender" shall appear (frame 1 to 50).
- Than Suzanne shall build up in 3 seconds (frame 51 to 150).
- Suzanne shall be clearly visible for a few frames (frame 151 to 200).
- After that the form shall dissolve and the particles move a bit until they vanish (frame 200 to 300)

105.1.1. Preparing the scene

- Remove the cube.
- Add a text object *Blender*. Name it *Blender-first*.
- Add a *Suzanne* (*Add->Mesh->Monkey*). Name it *Suzanne-first*.
- Convert the text object to a meshobject, because only mesh objects can emit particles (menu: *Object->Convert Object Type->Mesh*).
- Move the objects to their final positions, also the camera.
- **Clarification required - what are the final positions?** (**Noob note:** the author probably means to move the camera and objects to a right position for viewing the animation correctly.)

We need every visible object twice.

- Duplicate *Blender-first* and call the duplicate *Blender-last*.
- Duplicate *Suzanne-first* and call the duplicate *Suzanne-last*.

One problem is that the objects have different sizes. So how many particles shall be used to create a clearly visible form? Since the most difficult object is Suzanne, I have first chosen an acceptable representation and set everything else accordingly. But I will discuss the settings of the objects in order, and not in the way I did it in reality.

105.1.2. Set up the particle systems

Blender-first:

- Add a particle system to *Blender-first*.
 - Amount 20.000
 - *Sta:* 1, *End:* 50, *Life:* 50
- *Emit from:*
 - *Random/Faces/Even/Random*

Leave all other settings at default. This object will appear in the first 50 frames. After that the object has to vanish in the 51th frame.

- Insert a layer key for *Blender-first* in frame 50 (mouse over the 3D window, **I->Layer**).
- Change to frame 51.
- Move the object to layer 20 (**M->**select layer, bottom right box). Make that layer visible.
- Insert the next layer key for the object.
- Make layer 20 invisible again (select layer 1).
- Change back to frame 1.

Suzanne-first

- Add a particle system.

- Amount 20.000
- *Sta:* 50, *End:* 100, *Life:* 200 (we don't need this long life but the lifetime doesn't matter as long as it's long enough)
- *Emit from:*
 - *Random/Faces/Even/Random*
- *Visualization:* None

Leave all other settings as default. These particles shall not move and they shall not be visible. They are simply a target to the next system.

- Animate the layer of this object also. Move it out of sight in frame 151.

Blender-last

These particles shall move from the shape "Blender" to the "Suzanne" shape.

- Add a particle system to *Blender-last*.
 - Amount 20.000
 - *Sta:* 51, *End:* 51, *Life:* 100
- *Emit from:*
 - *Random/Faces/Even/Random*
- Physics:
 - *First/Keyed* It is the first system of a chain (a very short chain in this example) of systems.
 - *Keyed Target:* Suzanne-first, *Psys:* 1. Target is the first particle system of "Suzanne-first".
- Set time to *Absolute* in the *Extras* panel. We're going to animate the material of the particles, I want all particles to change at once in a certain frame.

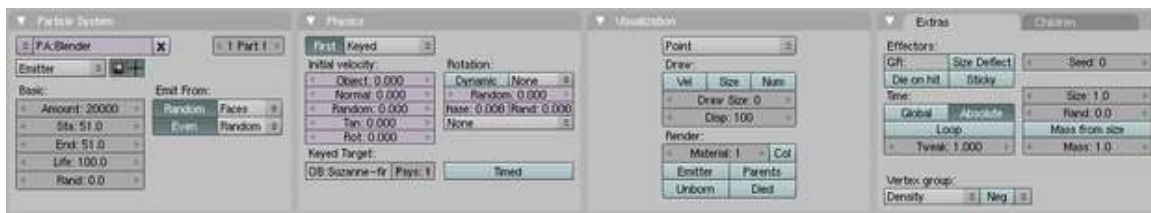


Figure 836 Image 2a: Particle system of the first keyed system.

Suzanne-last

These particles shall dissolve the form, so at first they take the shape and move around randomly afterwards.

- Add a particle system to *Suzanne-last*.
 - Amount 20.000
 - *Sta:* 151, *End:* 151, *Life:* 150, *Rand:* 0.5. The lifetime is randomized from 151/2 to 151 frames. [Noob Note: In 2.49a, you can't set the start later than the end, so you must first change the end to 151, then change the start.]
- *Emit from:*
 - *Random/Faces/Even/Random*
- *Bake:* End frame 300. Though we don't use baking here, you have to set the *End* frame in the *Bake* panel. Else the last 50 frames wouldn't be calculated.
- *Time:* *Absolute*

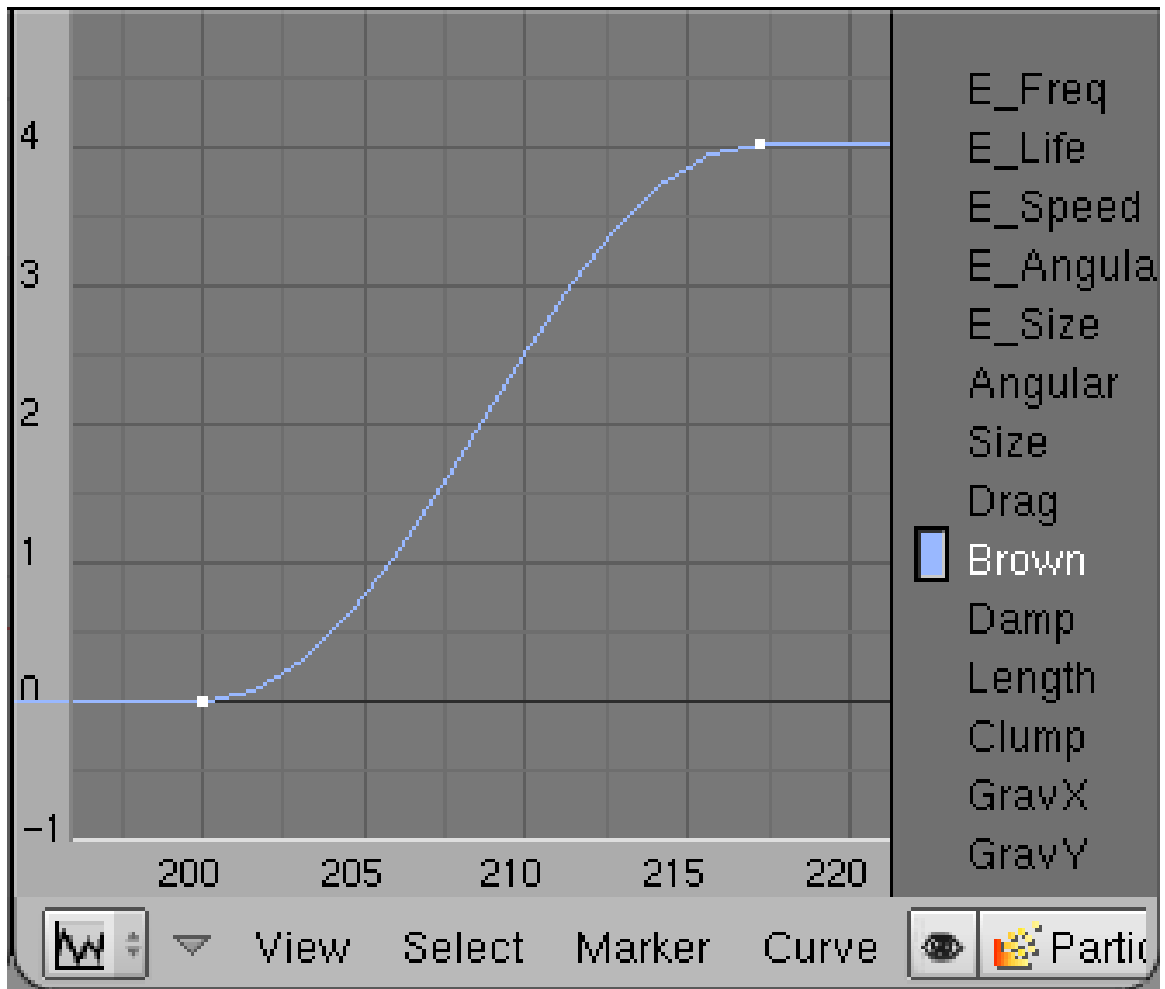


Figure 837 Image 2b: Animation of Brownian Motion

Animate the Brownian motion of the particles.

- Open an *Ipo* window.
- Select *Ipo Type* particles.
- Select the *Brown* channel (**LMB**).
- **Ctrl-Click** with the left mouse button in the *Ipo* window to create an Ipo-key for the selected channel.
- Create a curve similar to **Img. 2b**. **Brown** = 0 to frame 200, then increase to **Brown** = 4 in frame 217.

105.1.3. Material

Blender-first and Blender-last will get the same material, Suzanne-first is invisible anyway, but Suzanne-last will get a material. The material of the three objects is very similar, but I have animated the visibility of the particles to match the brightness of the objects.

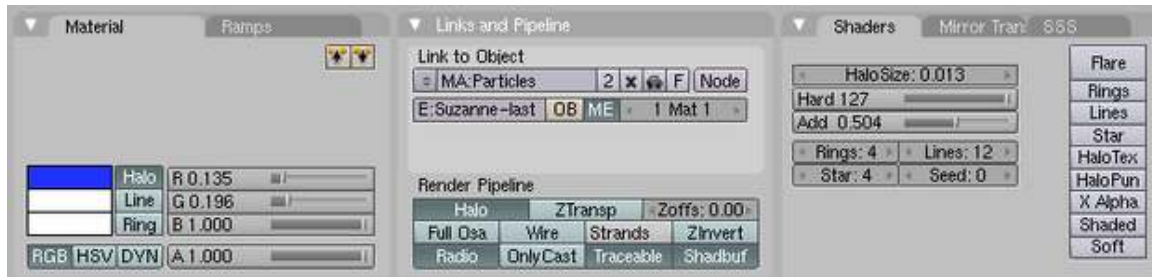


Figure 838 Image 2c: Material settings for the particles

Suzanne-last

- Create a simple *Halo* material like that in **Img. 2c**.

This is nothing special, relatively small particles with a relatively "sharp" edge. The *Add* parameter gives more brightness if particles overlap, I think this is a nice effect here.

Blender-first

- Assign the material also to "Blender-first".
- Make the material a *Single User Copy* by clicking on the small number next to the material name.
- Set *Alpha* to 0.438.

Blender-last

- Assign the "Blender-first" also to *Blender-last*".
- Make the material a *Single User Copy* by clicking on the small number next to the material name.
- Animate the alpha value of the material.
 - Insert an *Ipo* key for *Alpha* in frame 50.
 - Change to frame 150.
 - Set *Alpha* to 1.
 - Insert an *Ipo* key for *Alpha*.

Now the brightness of all systems are matched.

105.1.4. Links

- Download the blend file¹
- Manual about Particle physics²

1 http://www.snapdrive.net/files/180816/Blender3D_PartikelShapingObjects-2.48a.zip

2 http://wiki.blender.org/index.php/Doc:Manual/Physics/Particles/Particle_physics

105.2. Harmonic Force Fields

So here we will use *Harmonic* force fields. The Text "Blender" shall be transformed to the text "2.5" and than to the text "The big leap forward" (**Img. 1b**).

- Use mesh objects and let them emit particles. Only the particles from the first object are visible and move.
- The other particle systems use an animated *Harmonic* force field to attract the particles from the first system.
- The particles from the first system take the place of the other particles.
- Choose a good visualization of the particles and animate that also. This is difficult and maybe you should animate the visualization at last.
- At last the hard part: sync the different animations and force fields.

105.2.1. Preparing the scene

- Remove the cube.
- Add three text objects:
 - "Blender"
 - "2.5"
 - "The big leap forward"
- Convert the text objects to mesh objects (menu: *Object->Convert Object Type->Mesh*).

105.2.2. Set up the particle systems

- Give each of the three objects its own particle system.
- The first object "Blender" gets following system:
 - Type *Emitter*
 - *Amount*=4000. We need a lot of particles, to fill the last text evenly. If you want to make the animation even better, you could animate the *Halo* size of the particles to match the density of the different objects.
 - *Sta(rt)*=1, *End*=10, *Life*=250. The particles are created from the first to the 10th frame and live for 250 frames.

Only the particles of this system will be visible later.

- Change the calculation method from *Midpoint* to *RK4*. We need the extra precision here.

The objects "2.5" and "The big leap forward" will get particle systems with very similar settings:

- "2.5":
 - Type *Emitter*
 - *Amount*=4000
 - *Sta(rt)*=30, *End*=30, *Life*=70. So all the particles are emitted in frame 30 and live 70 frames. So their force field (which we give them soon) will start in frame 30 also. If you want a smoother transition you should give the particles a bit more time to be emitted (like from frame 30 to frame 40).
 - Emit from *Random/Faces/Even/Random*.

- Physics *None*. The target particles shall not move.
- Visualisation *None*. The target particles are not rendered.
- "The big leap forward": All settings as for "2.5", except *Sta*, *End* and *Life*.
 - *Sta*=101, *End*=101, *Life*=90.

105.2.3. The force fields



Figure 839 Image 3a: The object does not get a force field.

The **particle systems** of the objects "2.5" and "The big leap forward" will get a force field now. The objects **don't** get a force field.

- Select the object "2.5".
- Select the particle system in the *Field* panel of the *Physics* buttons (**Img. 3a**).

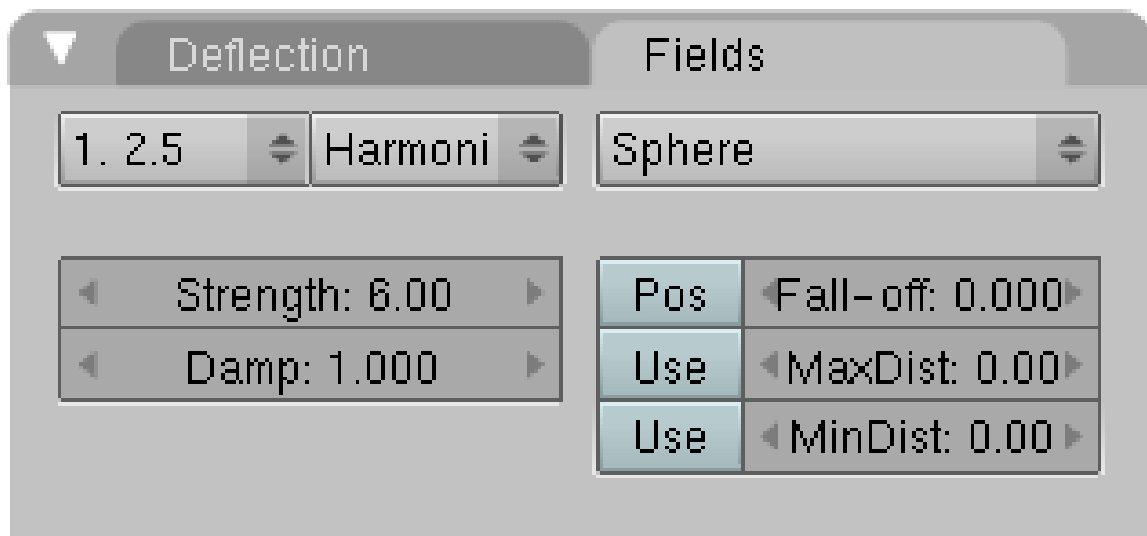


Figure 840 Image 3b: The particle system gets the force field.

- Use field type *Harmonic*. Now the particles from the object "2.5" do attract the particles from the object "Blender".

- The *Strength* of the force field determines the time the particles need to reach the target. Set it accordingly.
- The *Damp(ing)* makes sure, that the particles move to their target but not beyond.

Do the same for the object "The big leap ...", i.e. select the particle system and give it a *Harmonic* force field.

I have used a *Wind* field in the last frames to blow the particles away.

(Noobie Warning: When i followed the tutorials up to this point, i didn't know how to use wind. I thought selecting *Wind* instead of *Harmonic* would create the effect but then blender froze on me (I am guessing it went away and calculated wind for every single particle). so don't try wind yet. Wind is instructed in later tutorials.)

105.2.4. Links

- The Blend file to download³
- The manual about force fields⁴

³ http://www.snapdrive.net/files/180816/Blender3D_HarmonicFieldText.blend.zip

⁴ http://wiki.blender.org/index.php/Doc:Manual/Physics/Force_Fields

106. Billboard Animation

This tutorial was created with Blender v2.49

Billboard¹ visualization of particles - and especially their animation - is one of the more arcane concepts in Blender. We're going to shed some light on this. Billboard visualization is extremely powerful, everything that can be done with a halo can also be done with a billboard. But billboards are real objects, they are seen by raytracing, they appear behind transparent objects, they may have an arbitrary form and receive light and shadows. They are a bit more difficult to set up and take more render time and resources.

Billboards are aligned square planes. If you move an aligned billboard in a circle around an object, the billboards always faces the center of the object. The size of the billboard is set with the Size of the particle.

Texturing billboards is done by using uv coordinates that are generated automatically for them.

- The main thing to understand is that if the object doesn't have any UV Layers, you need to create at least one in the objects Editing buttons for any of these to work (**Img. 1a**).
- Moreover, material should be set to UV coordinates in the Map Input panel (**Img. 1b**).

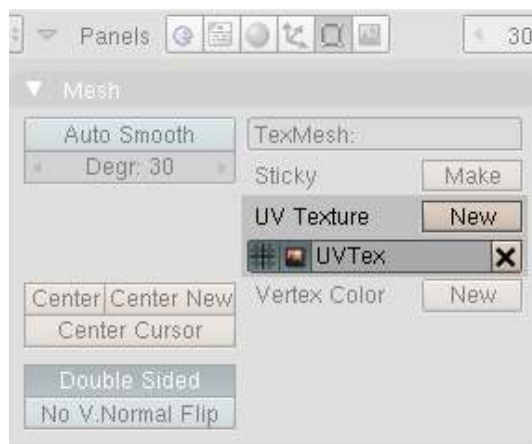


Figure 841 Image 1a: UV coordinates for Billboard textures

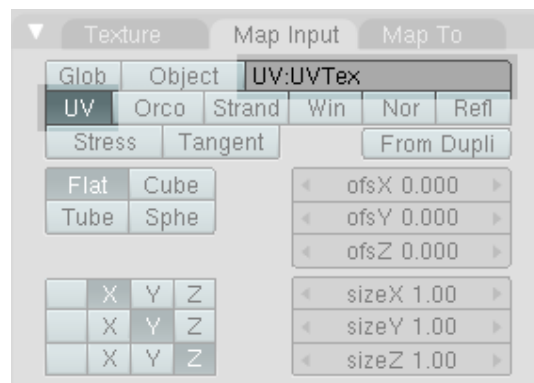


Figure 842 Image 1b: UV coordinates as Map Input

¹ <http://wiki.blender.org/index.php/Doc:Manual/Physics/Particles/Visualisation>

106.1. Splitting a texture

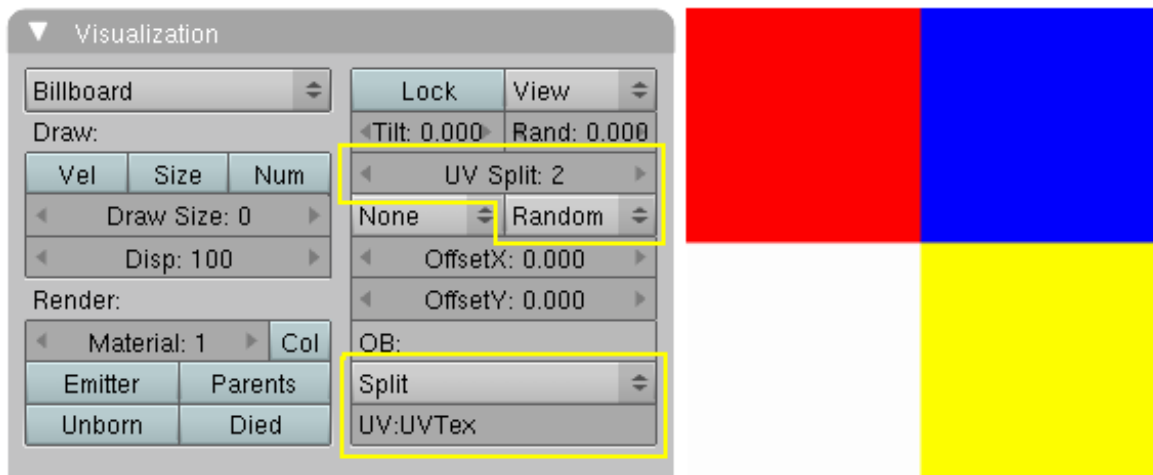


Figure 843 Image 2a: Billboard settings and texture for UV split

The simplest thing we can do is to give every billboard its own fraction of an image texture. We're going to use a simple image (**Img. 2a**, right hand side) and split this in four parts (*UV Split* = 2, horizontally and vertically).

- Create the UV coordinates (simply click on *New* next to *UV Texture* in the *Mesh* panel).
- Create a particle system.
- Activate *Billboard* visualization.
- Set *UV Split* to 2.
- *Offset: Random* to randomly choose one of the images sections. *None* would give only the first of the four fields, so all billboards would be red, linear would first use red, then blue, then white and then yellow repeatedly.
- Set *UV Channel* to *Split*. This creates the necessary UV coordinates, that are stored in the UV layer. If you use another UV channel, you need a second (or third) UV layer to store the coordinates (see below for an example).
- Fill in the field *UV* with the name of the coordinate set (here "UVTex").

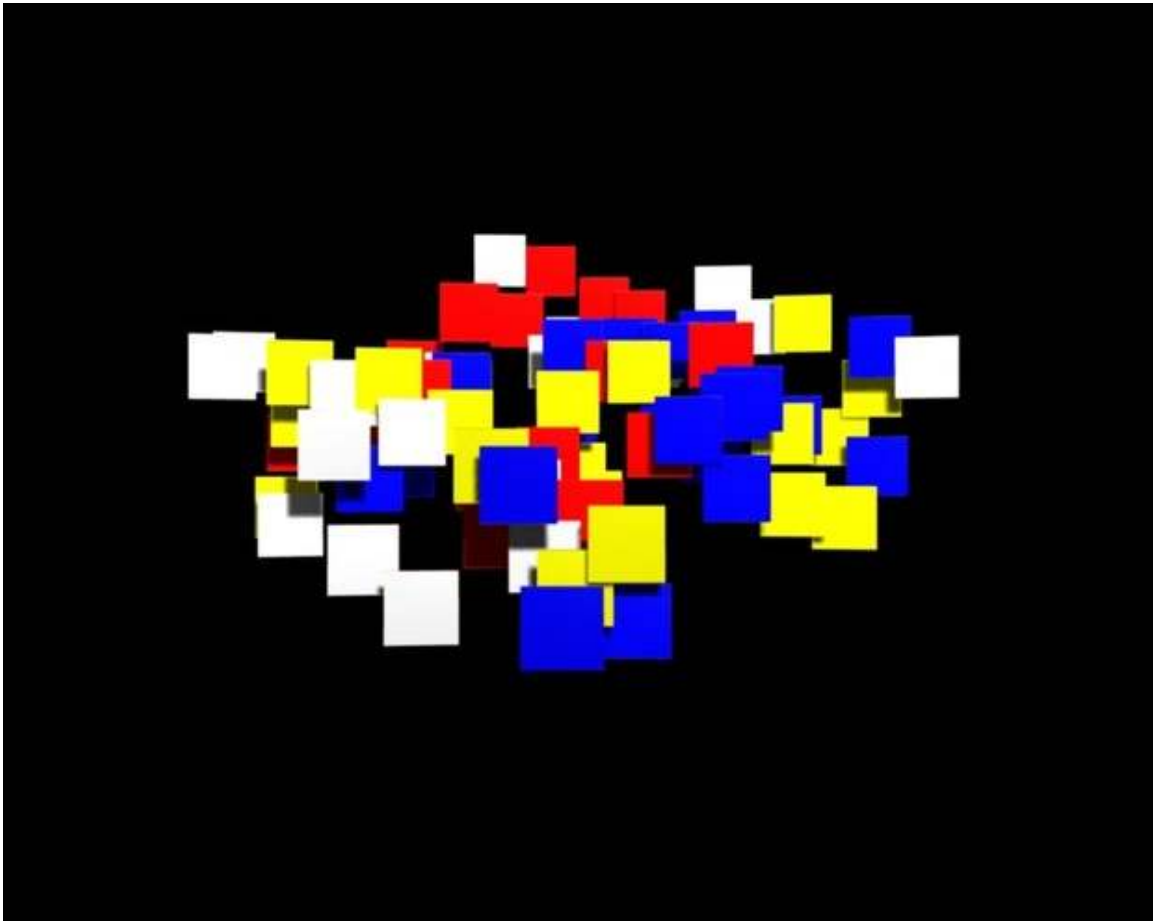


Figure 844 Image 2b: UV Split for Billboards Animation^a Blend file^b

a <http://www.vimeo.com/4116347>

b <http://www.snapdrive.net/files/180816/BillBoardSplitUVs.blend.zip>

- Now create a material for the particle emitter.
- Add a texture if not there already.
- Set *Map Input* to *UV*. If you only have one active *UV* layer this is used by default, but if you want to continue this tutorial you have to fill in the name of the *UV* set also (i.e. "UVTex").
- Load the texture as Image texture.

Now every billboard gets a random section of the texture (see **Img. 2b**).

106.2. Animating Alpha in relative particle time

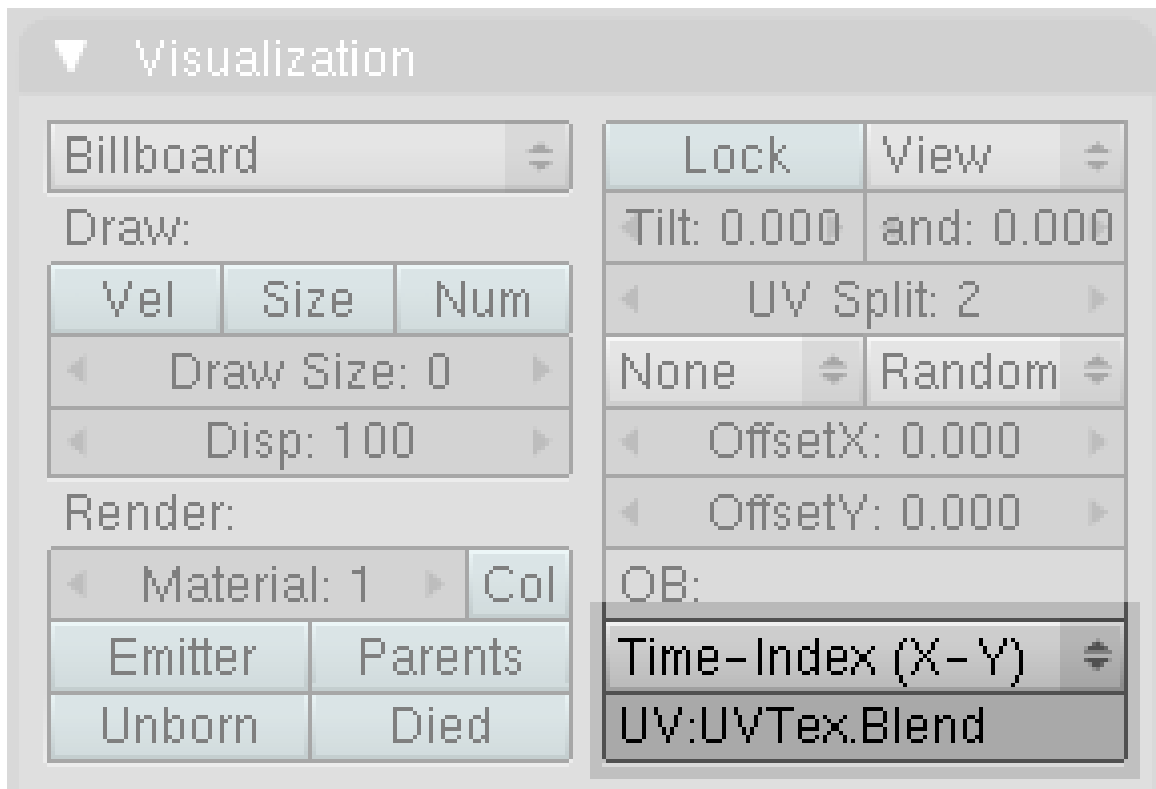


Figure 845 Image 3a: Using an UV layer for time animation

Now we're going to combine the texture split effect with an animation of the billboard alpha. This is done in *relative particle time*², i.e. in the lifetime of each particle.

Billboard time is setup completely different than particle time.

- Create a new UV layer (the second one). Name it "UVTex.Blend". Naming is not important here as long as it is unique.
- Select the UV channel *Time-Index (X-Y)* in the *Visualization* panel of the particle system. Fill in the name of the newly created UV layer "UVTex.Blend". This creates an additional set of UV coordinates in this UV layer.

Now both UV layers with different UV coordinates will be used. The material settings:

- Activate *RayTransp* and *TraShadow*.
- Add a second texture to your material.
- *Map Input* also *UV*, but now use the UV layer "UVTex.Blend".
- *Map To:*
 - *Invert Alpha* and *Invert Spec*
 - *No RGB* (since we're going to use a colorband texture)

² http://wiki.blender.org/index.php/Doc:Manual/Physics/Particles/Controlling_Emission%2C_Interaction_and_Time

- $DVar = 0$ to set Alpha to zero where the colorband is black.

This is one of the possible combinations for using a texture to make a material transparent (**Img. 3b**), there are several other methods that work as well.



Figure 846 Image 3b: Material for the *Alpha* animation. This is already the finished preview with the texture.

The texture is really the most important thing here.

- Use a linear *Blend* texture.
- Activate *Colorband* for full control.

Note:

The left hand side of the colorband is the first frame of the particle life time. The right hand side is the last frame of the particle life time.

- Adjust the colorband to your liking.

I've mapped the texture so, that "white" means full opacity, black full transparency.



Figure 847 Image 3c: Texture to control the animation time.

See this animation³ for the result of the combined textures. And also the Blend file⁴ if you like to take a look at it.

³ <http://www.vimeo.com/4121571>

⁴ <http://www.snapdrive.net/files/180816/BillBoardFadeUVs.blend.zip>

106.3. Animating billboard color

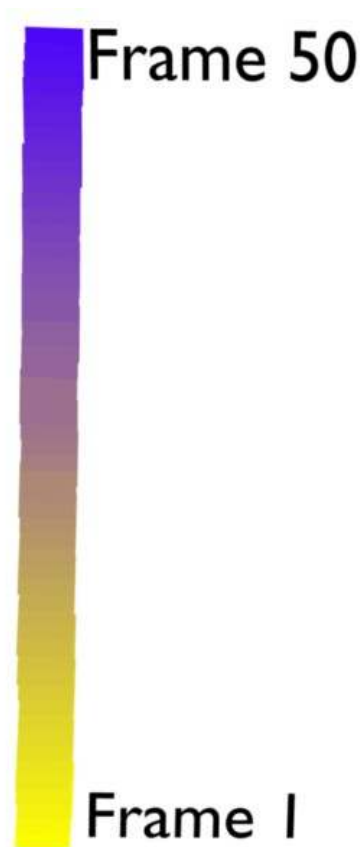


Figure 848 Image 4a: Animating Billboard color. Animation^a

^a <http://www.vimeo.com/4132553>

I will show here a second example for animating with a billboard texture, because this is so important. We're going to change the color during the lifetime of the billboard, but you can animate every property that way that can be influenced by a texture. Additionally you can animate the material itself, this is done in absolute time for billboards. So you can mix relative and absolute time animations.

- Open a new file.
- Remove the cube and add a plane.
- Add an UV layer to the plane
- Add a particle system with *Billboard* visualization and *UV Channel* Time-Index. Fill in the name of the UV layer.

The particle life time is 50 frames.

- Add a material to your plane.
 - Basis color *Yellow*.

- Use a linear blend texture. If you would use a color band here, you could create an even more colorful animation.
 - *Map Input UV*
 - Fill in the name of the UV layer.
 - *Map To Col.* I've used a dark blue color here.

Now the linear blend texture sets the amount of blue color that is mixed to the basis color of the material.

- At the left hand side the blend texture has a value of 0, so zero blue is mixed to the base color in the first frame of the particle lifetime.
- At the right hand side the blend texture has a value of 1, so 100% blue is mixed to the base color in the last frame of the particle lifetime.

106.4. Changing the starting color of a billboard



Figure 849 Image 4b: Flipped XY for the blend texture

If you flip the Blend texture, you get a different effect. Now every Billboard gets another starting color, but keeps that color during it's lifetime.

Have I mentioned that you may use multiple textures on a billboard, combining all effects?

107. Soft Body Animation

Due to rapid updates in Blender, some menu items may be different from those mentioned here. If the menus don't agree then just do what seems logical.

Softbody makes each individual vertex its own object that falls according to gravity and reacts to outside forces like fields. By adjusting the settings, you change the behaviour of the edges that connect the vertices. For example, you can make it so edges can stretch really far (aka elasticity), allowing the vertices to become distant, or you can make the edge stiff, so the vertices will always stay the same distance apart.

To put this in perspective, picture two cloths, one elastic and one cotton. The elastic one has edges that can extend, so if you view them in wire-frame (with vertices and edges visible) you would see the edges are more extended than an equal distance. The cotton one would only stretch a little bit, so the vertices would stay essentially the same distance apart.

We are going to make a big rubber ball, but not a big bouncy one, a flat (and somewhat lifeless) one. Start with a sphere. I would use a cube sphere or an icosphere, UV spheres don't deform well as they have too few vertices. A cube sphere is made by subdividing a cube and doing a "To Sphere" in the Edit window, under Mesh Tools.

Move the sphere up and place a plane below it. Make sure to do this in the right views so that it is aligned properly. Gravity acts on the z-axis (sphere should be above the plane relative to the z axis).

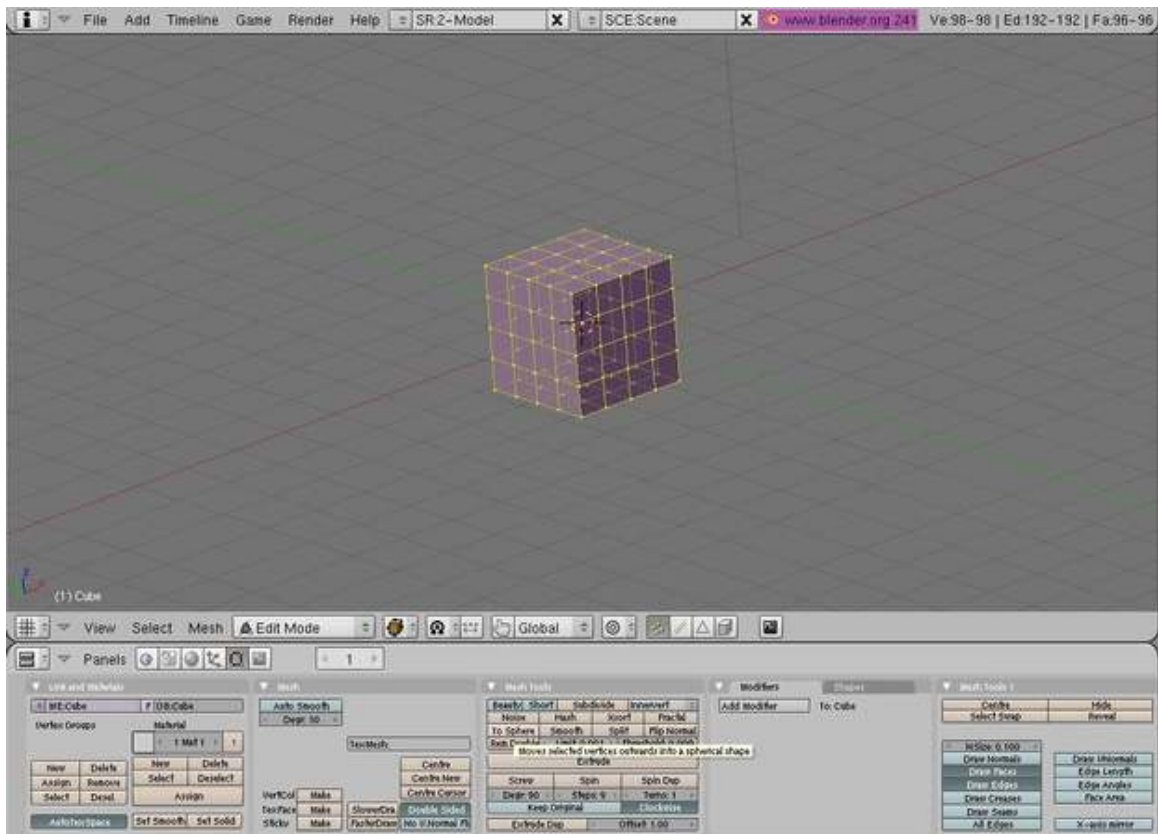


Figure 850

Now for the soft body select the ball and go to the object tab then the physics subtab or whatever your version has. Click "enable softbody" and then turn up 'Grav' to 9.8. Click off use goal. Press the > arrow key (a few times) and you should see the ball fall. The center will remain in place but this is not a problem. If you are on a slow machine you will notice lag. This is because blender moves it vert by vert, not efficient.

Note: in Blender 2.49, you must deselect the "Use Goal" button to release the center of the ball. Otherwise, it will just hang there.

Note: In Blender 2.44, click F7 on keyboard, select the "Physics Buttons" button, select "Soft Body".



Figure 851

When it reaches the plane it will pass through. To fix this we must make the plane affect the softbody. To do this make the plane deflect in its physics buttons.

(Noob Note: To do this in version 2.49 select the plane, go to object buttons -> physics context -> collision subcontext and select **Collision. The variables you can play with here are under the **Soft Body and Cloth Interaction**.)**--WithAlligators¹ (talk²) 10:00, 9 March 2010 (UTC)

¹ <http://en.wikibooks.org/wiki/User%3AWithAlligators>

² <http://en.wikibooks.org/wiki/User%20talk%3AWithAlligators>

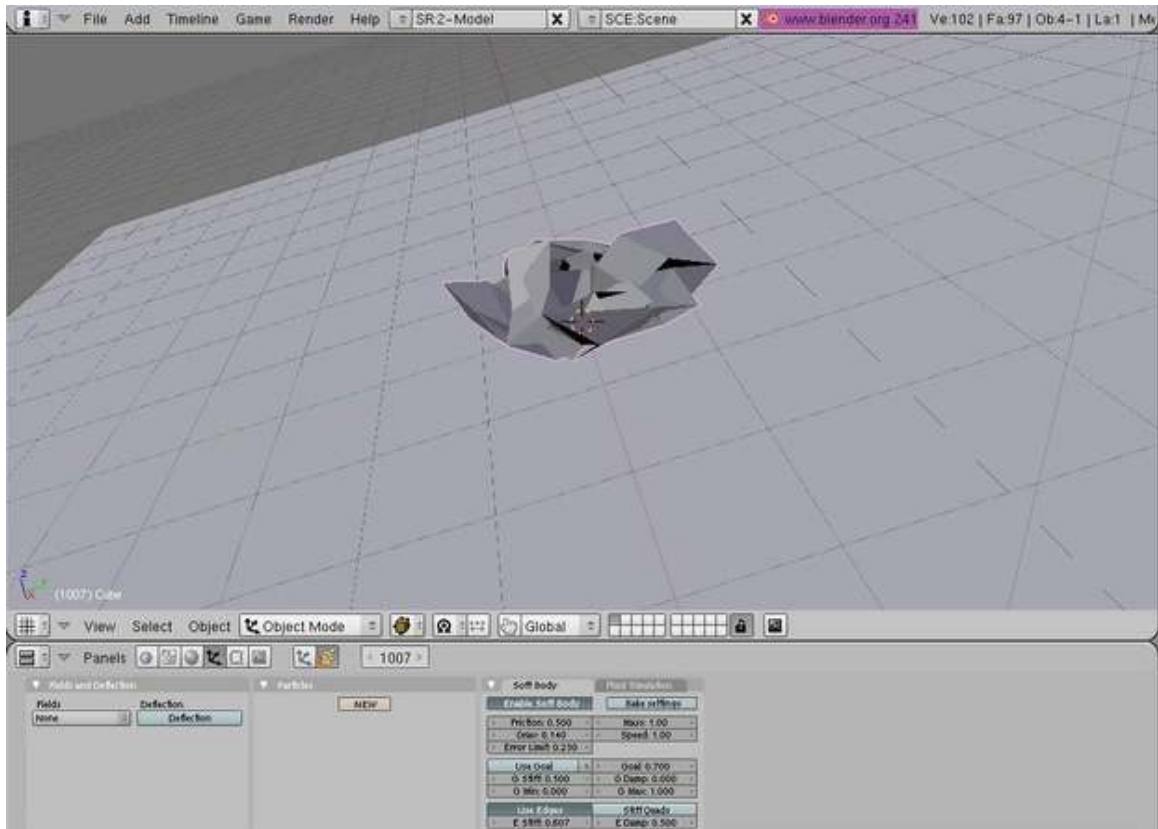


Figure 852

Now the ball collapses into a strange quivering wreck after impact. To fix this, you need to turn on the stiff quad button, but set the edge stiffness down a bit, so its more bouncy. You can use the bake function to solidify the settings see below

Note from Noob: *You might have to turn up the **Rigidity** Level to 0.100 (in the Soft Body Tab) as well in order to prevent the object from collapsing. (I used a subdivided cube as Object)*

*What I recommend to do before rendering as animation: In the **Bake Settings** (Soft Body Tab) Set **Interval** to 2 or 1, so the object will not start deforming too early before impact. This will slow down the bake process - just slightly - but make the object bounce more dynamically. Then bake again.*

*Noob note: In blender 2.46 you have to adjust the value of **Be** in the soft body tab, I changed it to 0.4*

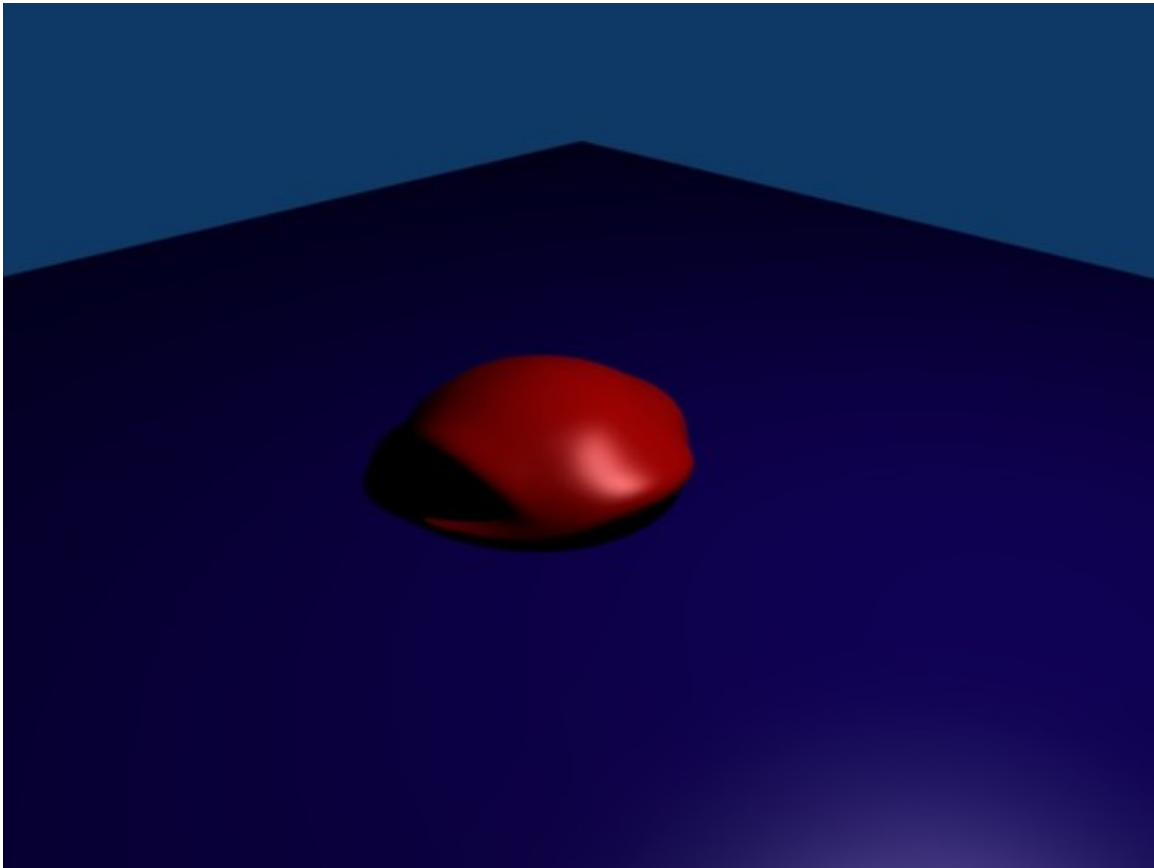


Figure 853

another thing you could do is animate the scene. Just put up animation video and set frames at about 50. Go to object mode, select sphere and press i key, and select LocRocScale. =)

107.1. Explanation of Settings

I invite you to correct and expand these definitions:

107.1.1. Softbody

- **Friction:** creates a resistance to movement of the whole object, like being submerged in a viscous fluid
- **Grav:** the rate of velocity change due to gravity. Results in a constant -z force.
- **Mass:** (Force = mass \times acceleration) affects everything by making the object heavier.
- **Speed:** tweaks the simulation to *run* faster or slower.
- **ErrorLimit:** raise it and the simulation will solve faster but strange things might happen. Save frequently, as blender might go nuts with this or any physic simulation (but less so after 2.4)
- **Goal:** makes the object try to return to its original position, useful at times, in the tutorial you could turn gravity off and key the ball falling and use this to keep it a ball.

- **use Edges:** uses the edges a means of resistance to movement for the object. Helps to keep it looking possible

107.1.2. Deflection

108. Simple Cloth Animation

In this tutorial, we will be making a simple skirt.

108.1. Making the Skirt Mesh

1. Open Blender and delete the default cube, if you aren't looking down on the scene, press **NUM7**.
2. **SPACE** → *Add* → *Mesh* → *Circle* (Blender 2.5 users press "Shift-A" to open the "Add" menu)
3. You should get a prompt for the number of vertices. Use about 12 and hit OK.
4. Once the circle is added, switch to Edit Mode with **TAB**. All the circle's vertices should be initially selected; if not, use **AKEY** to select them all.
5. Press **EKEY** and select "Only Edges" to extrude a second copy of the vertices; press **SKEY SHIFT+ZKEY** to scale the extruded vertices in the X and Y directions. These will make up the hem of the skirt; scale it out to as large as you like. Note that this is positioning the skirt out flat horizontally, instead of hanging down as you would expect; Blender's cloth animation system will take care of that, and this positioning gives maximum opportunity for the skirt fall in dramatic folds.
6. Now we will need to subdivide the mesh. The physics can only act on actual vertices, so the more of these we have, the more realistic the cloth effect will be. Select all vertices in the skirt, press **WKEY** and select the "Subdivide Multi" option; choose, say, 4 for the number of cuts, and hit OK.
7. While you're at it, go to the "Link and Materials" buttons, and click "Set Smooth".

108.2. Creating the Vertex Group

1. Now we have to specify that the waist of the skirt will stay fixed in place as it falls: deselect all vertices, and select the innermost ring of vertices. The easiest way to do this is select two consecutive vertices in that ring, then press **CTRL+EKEY** and use the "Edge Loop Select" option. (Shortcut: Alt + Click edge to select closest loop) With those vertices selected, go to the Editing buttons, in the "Link and Materials" miniwindow, under "Vertex Groups", click "New", perhaps give it a more meaningful name (like "Waist"), and click "Assign" to put the selected vertices into the new group.

108.3. Animating the Skirt

1. Tab out of Edit Mode into Object Mode. The skirt should still be selected.

2. Switch the Buttons Window to Object Context (**F7**), and select the Physics Buttons (second in the row of three icons that appears to the right of the context buttons).
3. Among the miniwindows should be one titled “Cloth”. Click the “Cloth” button in it; a whole lot of other settings should appear, most of which can be left at their default values. However, click the button titled “Pinning of cloth”; this will make a popup menu of the available vertex groups in the object. Since you only created one vertex group (the “Waist” one), this should already be selected by default.
4. Now the magic happens ... rotate the view to an oblique one to give yourself a good view of the process, and hit **ALT+AKEY**. You should now see the skirt fall from its horizontal position to a more natural vertical one, developing some folds in the process.
5. After the animation has run through at least one complete cycle, hit **ESC** to stop it.

108.4. Keeping the Folds

1. Now use the left- and right- arrow keys to step through the animation one frame at a time, until you find a position for the skirt that you like.
2. When the 3D view is showing a nice shape for the skirt, go to the “Object” menu, and look for the “Scripts” submenu. In here there should be a script called “Apply Deformation”. Select it.
3. It looks like nothing has happened, but in fact you now have a second copy of the skirt mesh, “baked” into the position corresponding to the current frame of the animation. Try using the arrow keys to move through the animation, and you will see the baked copy remain in the chosen position.
4. At this point, you can delete the original animated skirt mesh (or move it to another layer for future reuse), leaving the nicely-folded copy.



Figure 854

108.5. Extra Practice

1. You will notice at some points during the animation, the folds of cloth pass right *through* each other, which is of course impossible with real cloth. To prevent this, you could go to the “Collision” tab in the “Cloth” miniwindow, and click the “Enable selfcollisions” button. Rerun the animation (**ALT+AKEY**) to see the difference; what other effects does it have?

2. Maybe the folds don't look realistic enough. Go back to the original mesh, bring up "Subdivide Multi" again and subdivide it by a couple more levels. Rerun the animation (**ALT+A**KEY). It should take a bit longer for the first cycle, but do the results look better?
3. This YouTube tutorial might also help: <http://au.youtube.com/watch?v=mgYhZ3hWwTQ>¹ happy animating!

¹ <http://au.youtube.com/watch?v=mgYhZ3hWwTQ>

109. Soft Body with wind

This is the **Blender wind and soft body tutorial**. This tutorial will try and help your knowledge of using Soft Body and the Fields and Deflection panels in Blender 3D. (For best results, I recommend you use Blender version 2.43 or higher). Don't forget to save your work at various points throughout the tutorial.

109.1. Setting up scene

109.1.1. The plane

Delete the default cube [X key, or yes, even the Delete key] and enter Front view mode [Num. Pad 1]. Add a plane, scale 3 times [S key, 3, Enter] and sub-divide 4 times until you get something like the shape below. If your computer can handle more, and you want more, sub-divide as many times as you like, but if your PC is struggling with this, undo once or twice.

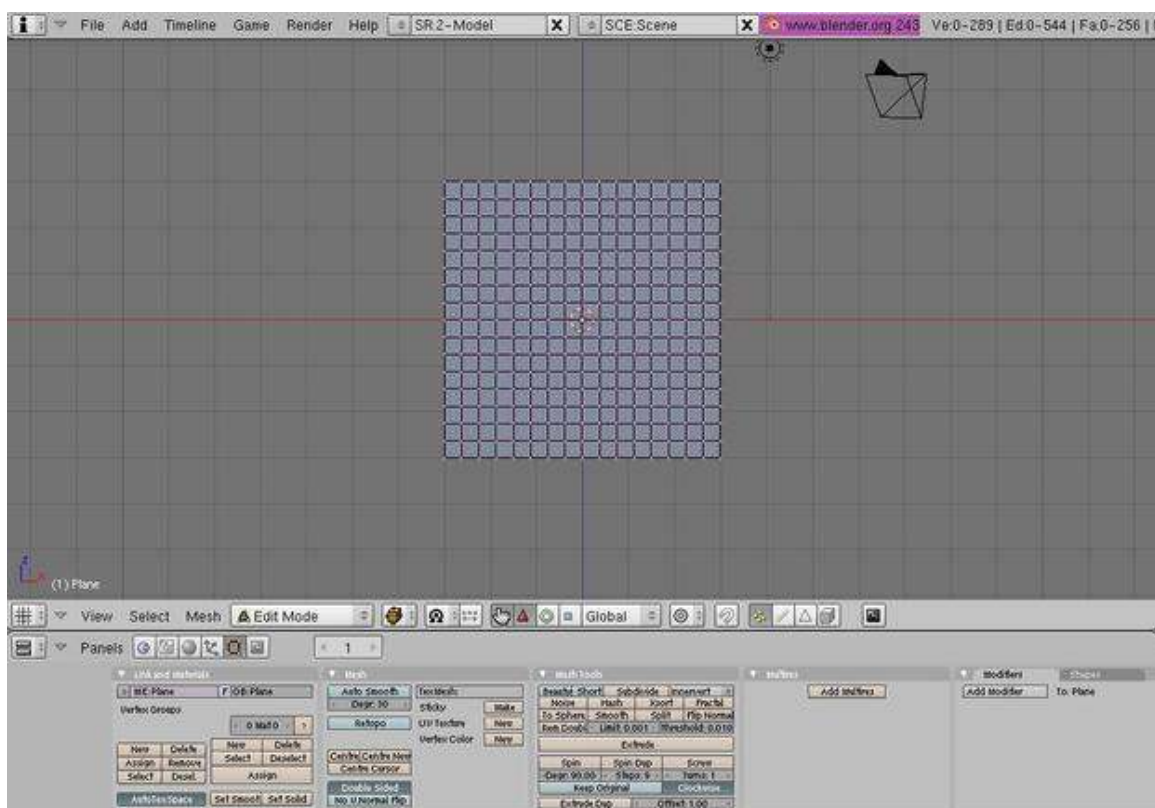


Figure 855

109.1.2. The wind items

TAB out of edit mode (if you haven't done so already), enter side view [Num Pad 3] and add an empty. Clear the rotation [ALT + R] and rotate it 90° [R, 90]. Place it about -5 from the centre [G, y, -5] and make sure its still in the middle of the X axis. Your scene should look like the picture below:

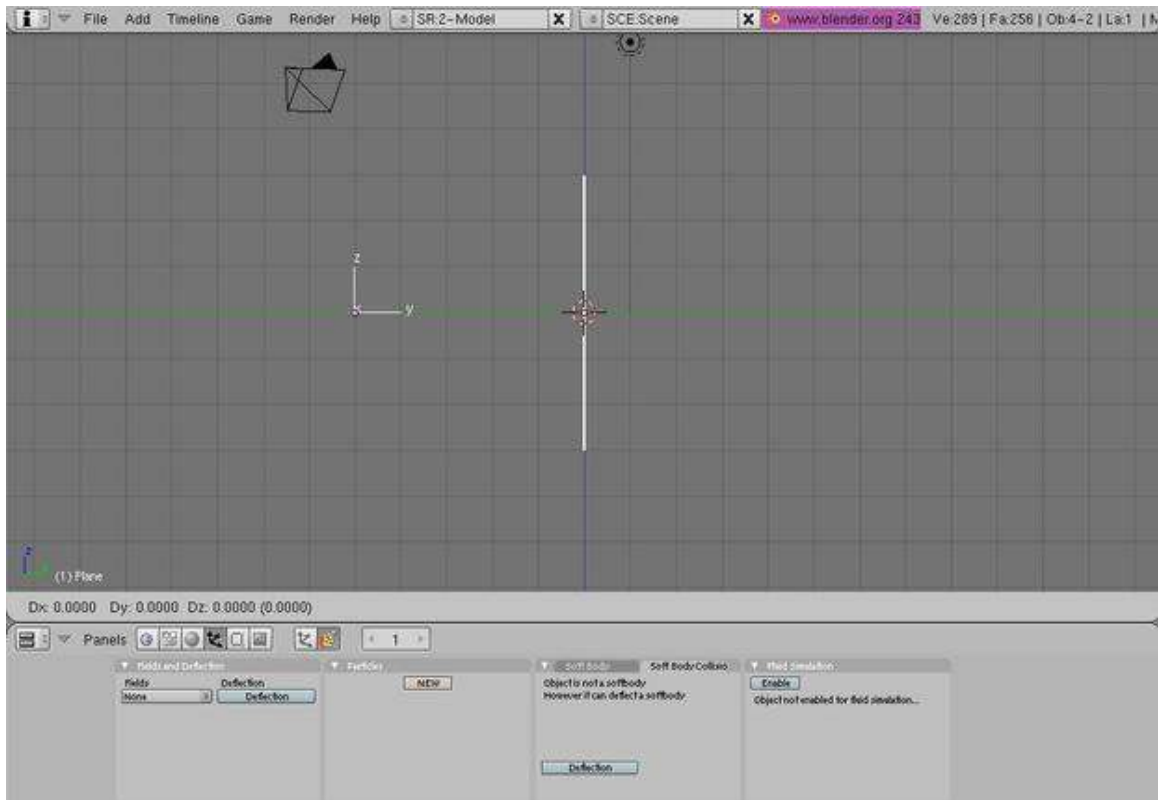


Figure 856

For the second wind item, add an empty like before and clear its rotation, but DON'T rotate it any more. Move it along the Y axis by 4 places [G, Y, 4] and -10 by the Z [G, Z, -10]. And now your scene should look something like this (with all items selected).

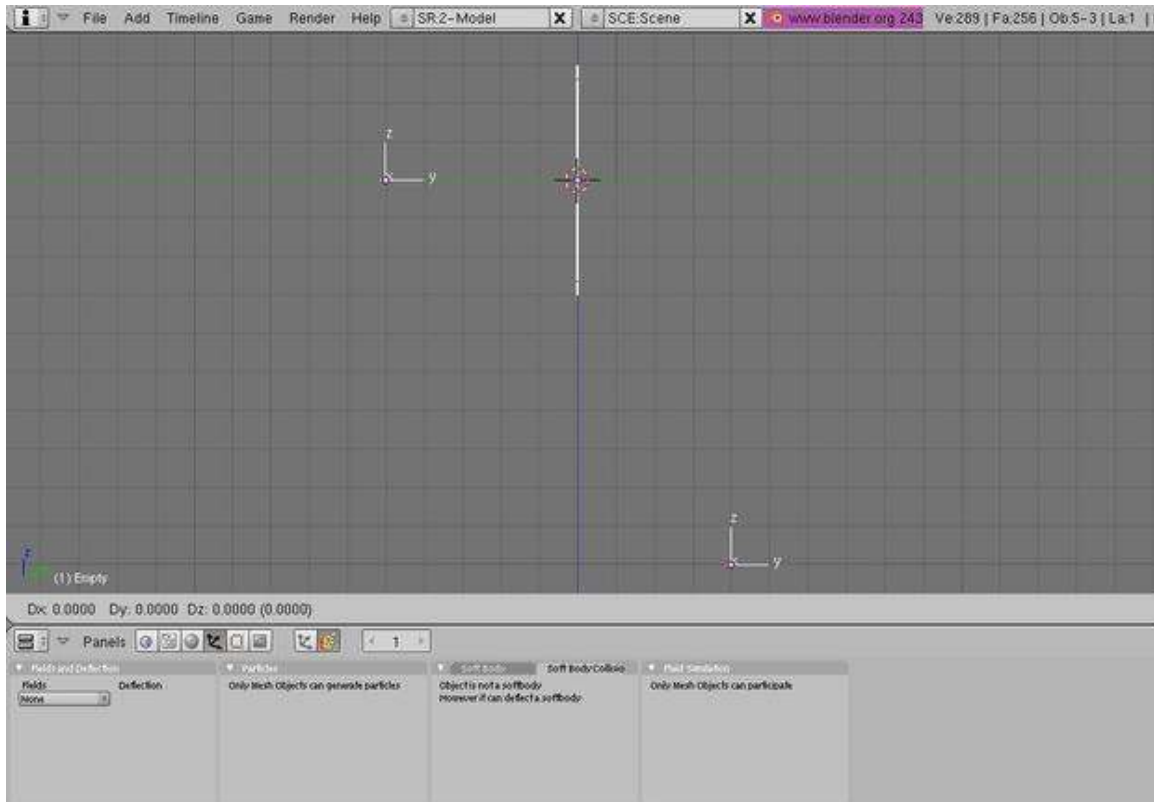


Figure 857

And that's all your items you'll need for this. Onto the harder part.

109.2. Designating each object's job

109.2.1. The plane

Now, the plane will become a soft body, so enter the Object Panel [F7], click on the Physics Button (beside the three arrows, see second picture below) and click on the soft body button. You will get a load of boxes, but we only need to look at the Grav, Goal, and the bottom parts:



Figure 858

Change the values of these as shown above. Change the Aero part to 732 (if you go up to 1000, it changes the final output by a bit).

Now, click on the Soft Body Collision tab and click on the Self Collision and Deflection buttons. You don't need to change any of the numbers here. The buttons window should look something like this:

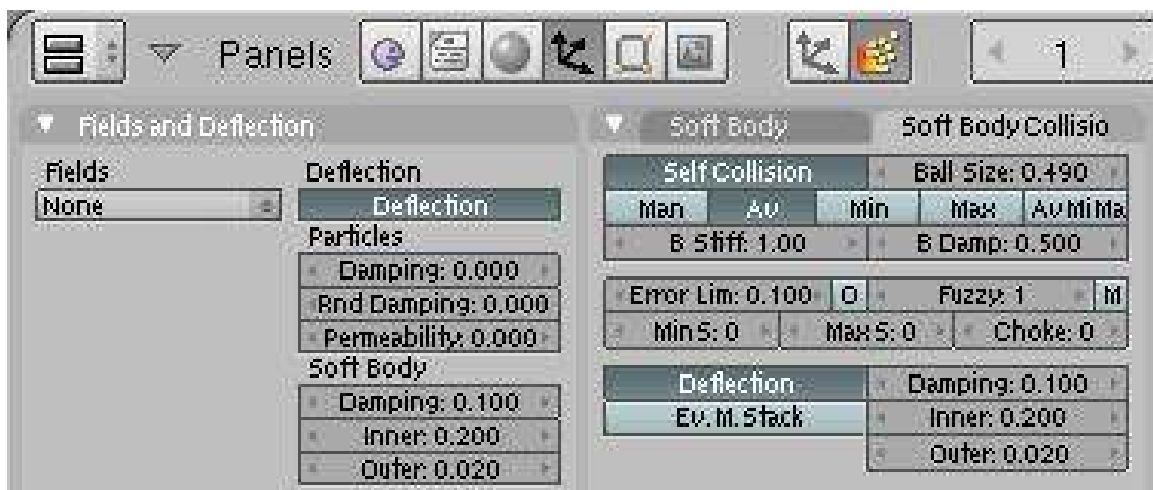


Figure 859

Now we will move on to the Empty items which will be the wind.

109.2.2. Creating the wind

The two empty's we created earlier will be the items that will act like wind, and when you render at the end, they will be invisible, just like the wind. Anyway, select one of the empty's and enter the Physics panel. Click on the box under the Fields and select "Wind". (Blender 2.5+ note: First you will have to choose "Force Field", then change the Type to "Wind".) Change the Strength to 20, click on Use MaxDist and change MaxDist to 2.5.

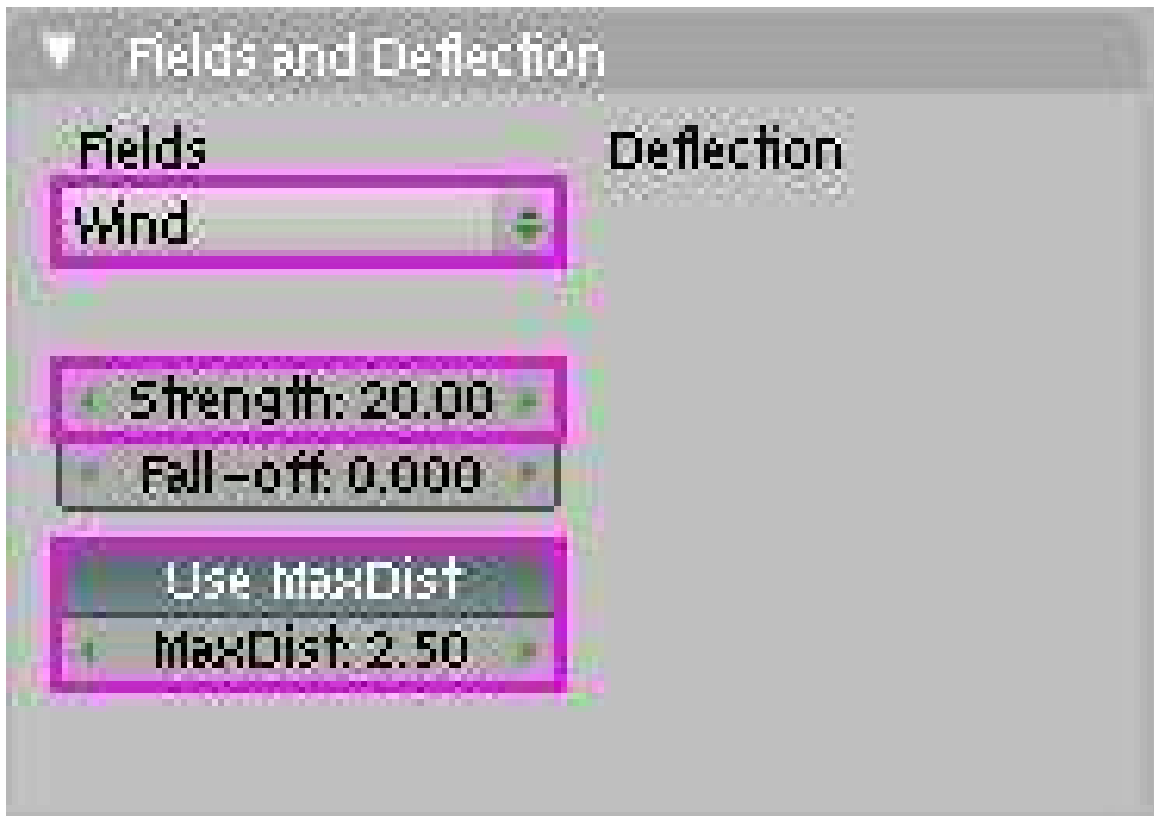


Figure 860

Do the same for the other empty. Your scene should look something like this:

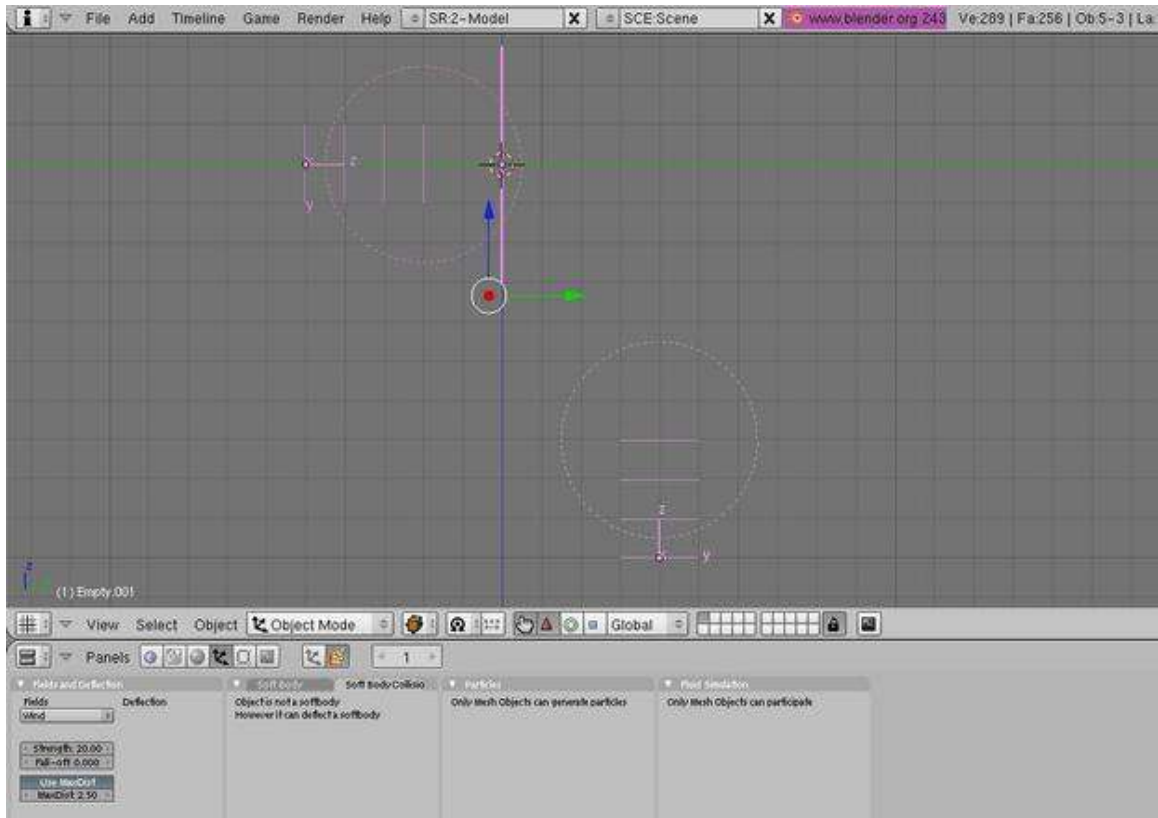


Figure 861

109.3. The movement

This is probably the most complicated part, but shouldn't take long.

109.3.1. The Empty on the left

In side view [Num pad 3], select the empty on the left and do the following: 1. Press [I key] and then select LocRotScale from the list it gives you. This inserts a key frame, saying that on this frame, keep the Location, Rotation and scale the same throughout the rest of the movie, unless it comes across another key frame, which might tell it to move or stay the same. Now, press the Up Arrow Key until you get to frame 31 (about 3 times). If you don't know where the box is that tells you what frame your on, check the following picture:

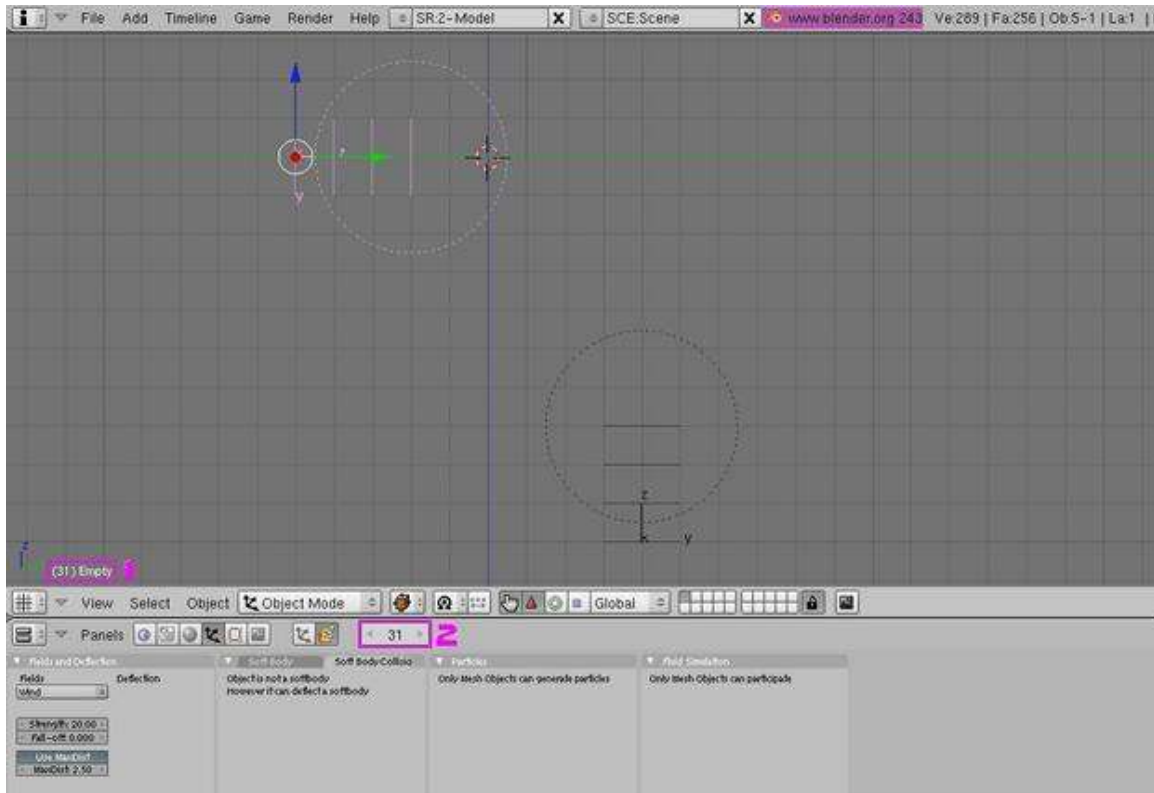


Figure 862

Number 1 shows the current frame number and the name of the selected object while number 2 shows the current frame and allows you to change the frame with the arrows on either side.

109.3.2. Back to the movement

On frame 31, select the empty on the left (the one selected in the picture above) and press [G key]. Now press [Y key] and press “4”, then [Enter key]. Now press [I key] and select LocRotScale again. Now, if you press and hold the Left Arrow Key, you should see the Empty move back to its starting place. Now, to check the curve guide (called the "IPO Curve Editor"); right-click on the top panel (number 1 in picture below), then select "Split Area" and click when the line is a bit out from the side (number 2 in picture below).

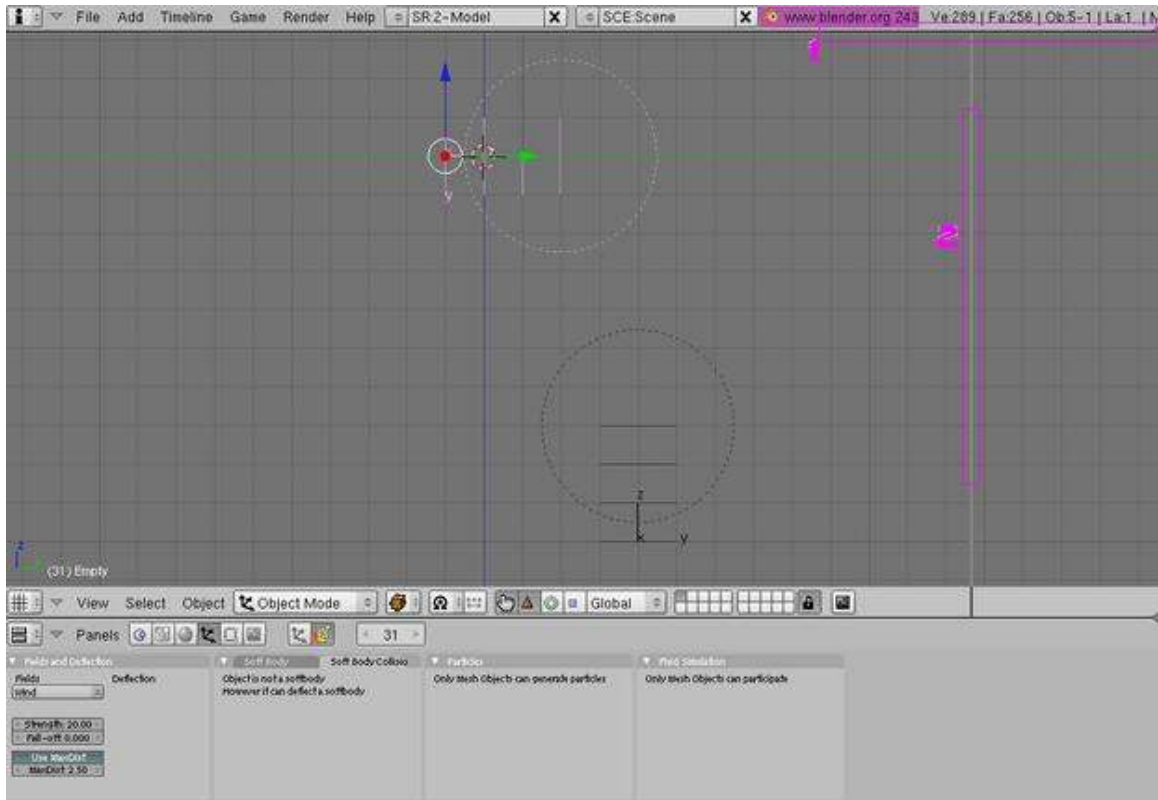


Figure 863

Now, click on the box (shown below) and select "IPO Curve Editor" from the menu.

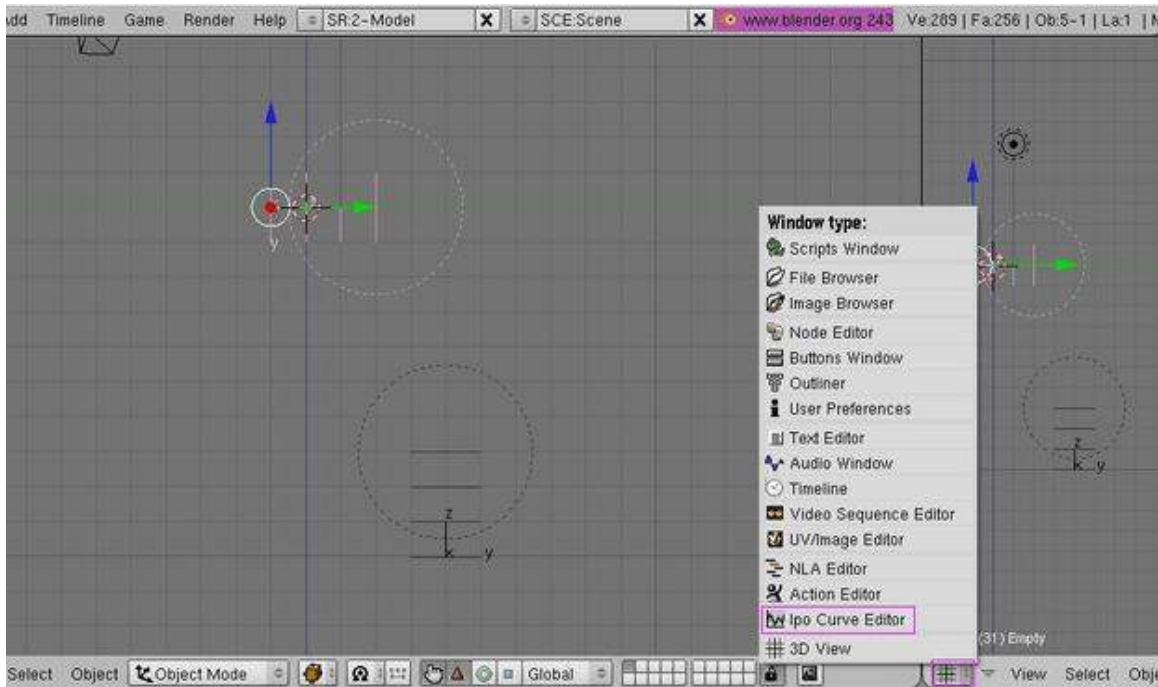


Figure 864

This window just shows where the key frames you entered are, and how much the shape changes, moves or rotates. Now, go back to frame 1 and press [ALT]+[A], which plays the animation. You should see the "wind" hit the sheet and the sheet will blow away. Pretty nice, huh? Now, it would be fine like that (I'll show you how to get rid of the blockiness in the last part), but say you want to hit the sheet again and send it upwards? We will use the second empty for this.

Select the second empty (the one on the bottom). Insert a key frame ([I key], LocRotScale) on frame 1 and frame 50. Go to frame 60, grab the empty and move it up 8 places ([G key], [Z key], 8). Insert a key frame ([I key], LocRotScale) and then go back to frame 1. You have completed the basics, and the most of this tutorial. Now, press [ALT]+[A key] and watch your animation.

109.4. Finishing touches

109.4.1. Blockiness

Now, to sort out the blockiness, go to the Editing buttons panel [F9] and select the "Set Smooth" button.

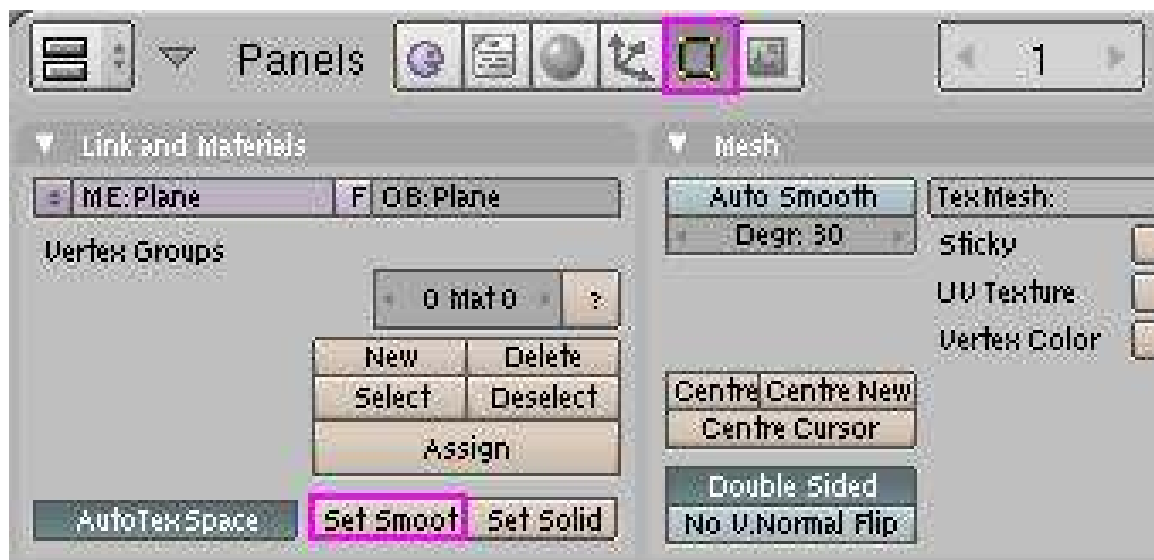


Figure 865

Now if you preview again ([ALT]+[A]) you should see the plane is smoother, but might be a bit unrealistic (i.e. not acting like real cloth). Now, that can't be helped, the only way to get rid of it is to subdivide the plane several more times, but that will put a LOT of strain on your computer, so I wouldn't recommend doing it unless your computer can handle it.

109.4.2. Background

First of all, got to the Scene panel [F10] and over at the Format panel, change SizeX to 400 and SizeY to 300. Now, to change the background colour when you render, go to the shading panel [F5]

and select the World buttons. Now, in the World panel change HoR to 0, HoG to 0.80 and HoB to 1.0.



Figure 866

All that does is just changes the background colour to a brighter, nicer shade of blue. Now, I'm just getting a bit lazy, so I'll just explain this part with a picture: (don't forget to select the plane).



Figure 867

Change the values and press the buttons shown in the picture. This will change the colour of the plane to white.

The only problem now is the light. Press [F12] and you should see what I mean. There's no light shining on the plane where the camera is, so its just shaded dark. To fix that, select the light and go to the shading panel [F5]. You now have options for the light. Click on the "Sun" button and press [ALT]+[R] to clear rotation. Now, in side view [Num pad 3] press [R] and then type -45. Enter top view [Num pad 7] and press [R] and then -45. This now shines the light in the direction of the plane, but, if like mine, the light is above the plane in top-view, press [G], [Y] and then -5. The line

coming out of the light source should be roughly pointing at the centre of the plane. Here's what your window should look like:

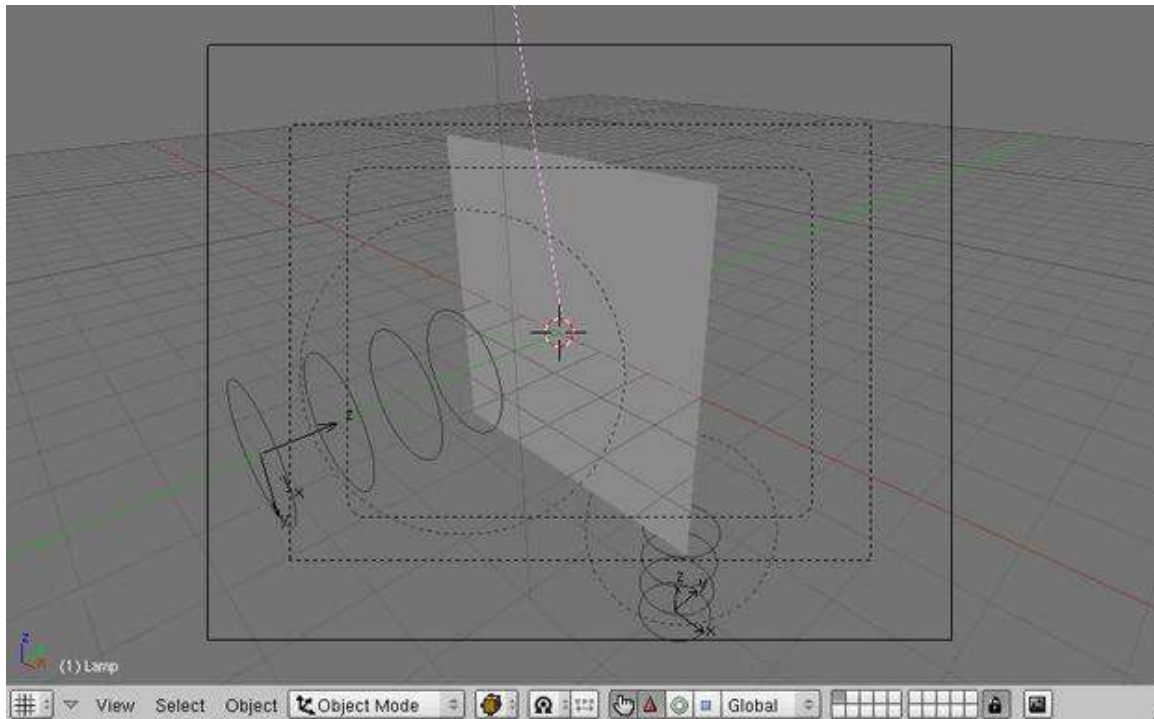


Figure 868

The pink line is the light. This was taken in camera view [Num pad 0]. Render the frame and you should see what the final product looks like. Move to any frame you want and press [F12] and it will show the rendered image for that frame. Now, rendering the entire animation might, in fact, will take a long time unless you have a completely new PC, are using a big server like DreamWorks (one of the big PC's, used for rendering images faster) or have a quicker renderer, I don't use any external renderers, and am stuck using a PC that is about 2 years old (in other words, full of junk). It took my PC less than 5 minutes to render 100 frames, and the last 25 were just blank (as the plane just flew above the camera). To change the amount of frames to render, and to render the full animation, here's what you press:



Figure 869

The box that says End:100 lets you select what frame you finish rendering the animation on (NOTE: the Sta:1 button lets you select what frame to start rendering the animation). When you press the

ANIM button, it starts rendering the animation. The Box covered in blue, that says Jpeg, lets you change the format of the image in the end.

To export the animation as an AVI format, select the AVI button shown below, then press OK. Now, you need to Animate the sequence again, so press the ANIM button again.

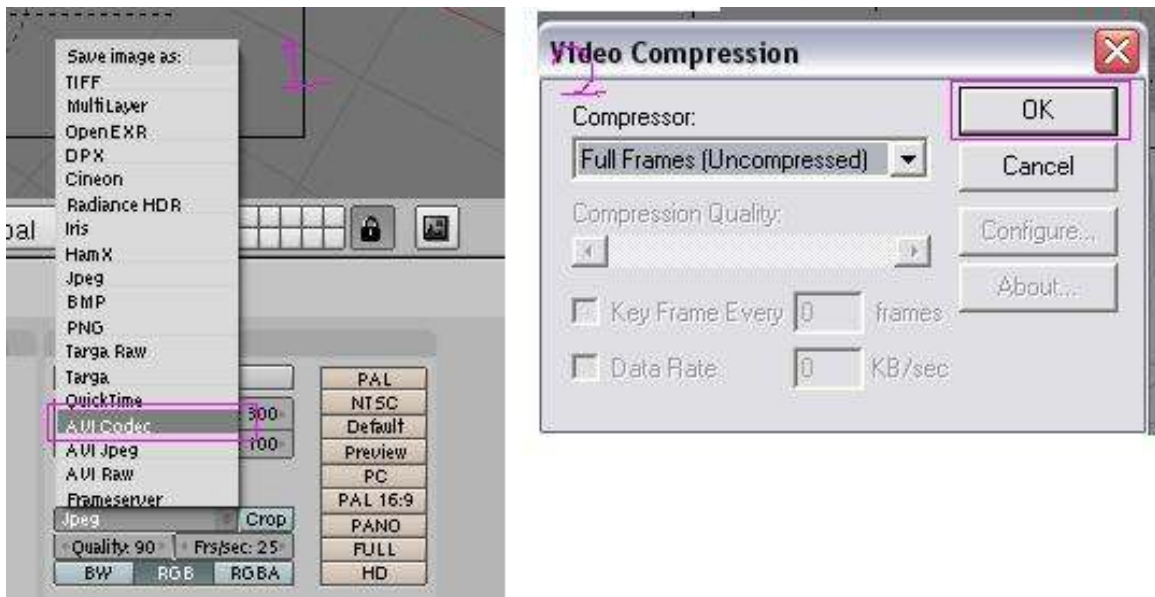


Figure 870

The finished .AVI will be saved in a folder with the rest of your rendered images. It should be saved in a folder called tmp, which (on a Microsoft computer) is in My Computer/tmp. The name of the file will be whatever your start and end frames are, so in this movie, mine is called 0001_0100.avi. You can now do whatever you want with your movie now, and hopefully you have learned a trick or two that will be helpful later on in your Blender career.

109.5. Final result

Here is an animated .gif of my final result: (imported into Macromedia Flash and exported as an animated .gif) You are required to be an auto confirmed user or uploader.



Figure 871

--User:10RCR¹ /

¹ <http://en.wikibooks.org/wiki/User%3A10RCR>

110. Your First Test

The Blender Game Engine is an interesting feature of Blender. It is basically a 3D environment in which 3D objects move around and react to each other upon contact. One common application is to recreate 3D architectural tours.

In this tutorial, you will learn the basics of object collision within the Blender Game Engine (BGE). From Blender games to use in animations, the bullet physics engine offers a massive number of possibilities. The tutorials found within this wikibook on the subject of the BGE are generally focused on game creation, but the concepts taught within can be applied to a multitude of situations.

As a start, we will teach you to make a ball roll realistically down the hill using Blender's game engine.

110.1. Adding the Hill

First, make a plane, then switch to Edit mode (**TAB**), and multi-subdivide it with 2 cuts (**WKEY** → *Subdivide Multi* → 2). Next, enter face select mode (**CTRL+TAB** → *Faces*) to drag the center face up, in order to form a rough outline of a hill. Add a subsurf modifier (in the edit buttons) to about 3, then apply. You should now have a serviceable, but small, hill. Scale the hill up (**SKEY**) by about 10 times, and we're ready to add the ball.

Noob note:

1. You can also use a 3*3 grid.
2. Delete the original cube first.
3. The subsurf modifier is not essential.

110.2. Creating the Ball

Now, add an icosphere (**SPACE** → *Add* → *Icosphere*) and relocate it to be just above the hilltop (**GKEY** or use the translate widget by pressing **CTR+ALT+G**). Let's change the color of the sphere so we can differentiate it from the plane. Go to the material buttons (with the sphere selected) and click on the white panel beside the COL value. In the color selection widget that appears, change its color to a bright red.

Next we need to make the Physics engine iterate over it. With the sphere selected, go to the logic buttons (the little purple Pacman-icon). **NOTE: In Blender 2.5 and above the Logic Buttons are gone. In order to have the 'Actor' button, click on the button showing 'Blender Render', and select 'Blender Game' Engine. Then go to the physics tab in the buttons menu. There you will see the 'Actor' Button.** You will see a button in the top left that says *Actor*. Press it. Now select

from the selection box beside of the "Actor" button *Rigid Body*. This makes the ball roll, instead of staying completely upright the entire time. You will see a bunch of settings available now. Change *Radius* to 2. This changes size the physics engine *thinks* the ball is. You notice a dotted circle around the object; this is the boundary. Now change the *Radius* back to 1. You now have your first Blender game ready to go.

Noob Note:

1. Make sure you are in object mode first before you add any object.
2. **F4KEY** is the shortcut to the logic panel.

110.3. Testing your game

Now the time has come for the first test of our game.

1. Add a light source well above the hill (**SHIFT+AKEY** → *Lamp* → *Lamp*). Align in front view (**NUM1**)
2. Press **NUM5** to switch to Perspective mode, which gives a realistic view, rather than a view in which objects stay the same size with distance (be sure to switch back to Orthographic view when you are editing again using **NUM5**)
3. Enter textured mode (**ALT+ZKEY** -- press **ZKEY** to switch back to solid view mode)
4. Switch into side view (**NUM3**) and press **NUM8** several times to get a good perspective on the ball.
5. Press **PKEY** to play the game (Make sure you are in Object mode (**TAB**))
6. Press **P** to start testing the game. You should see the red ball drop onto the hill.
7. Press **ESC** to quit testing the game

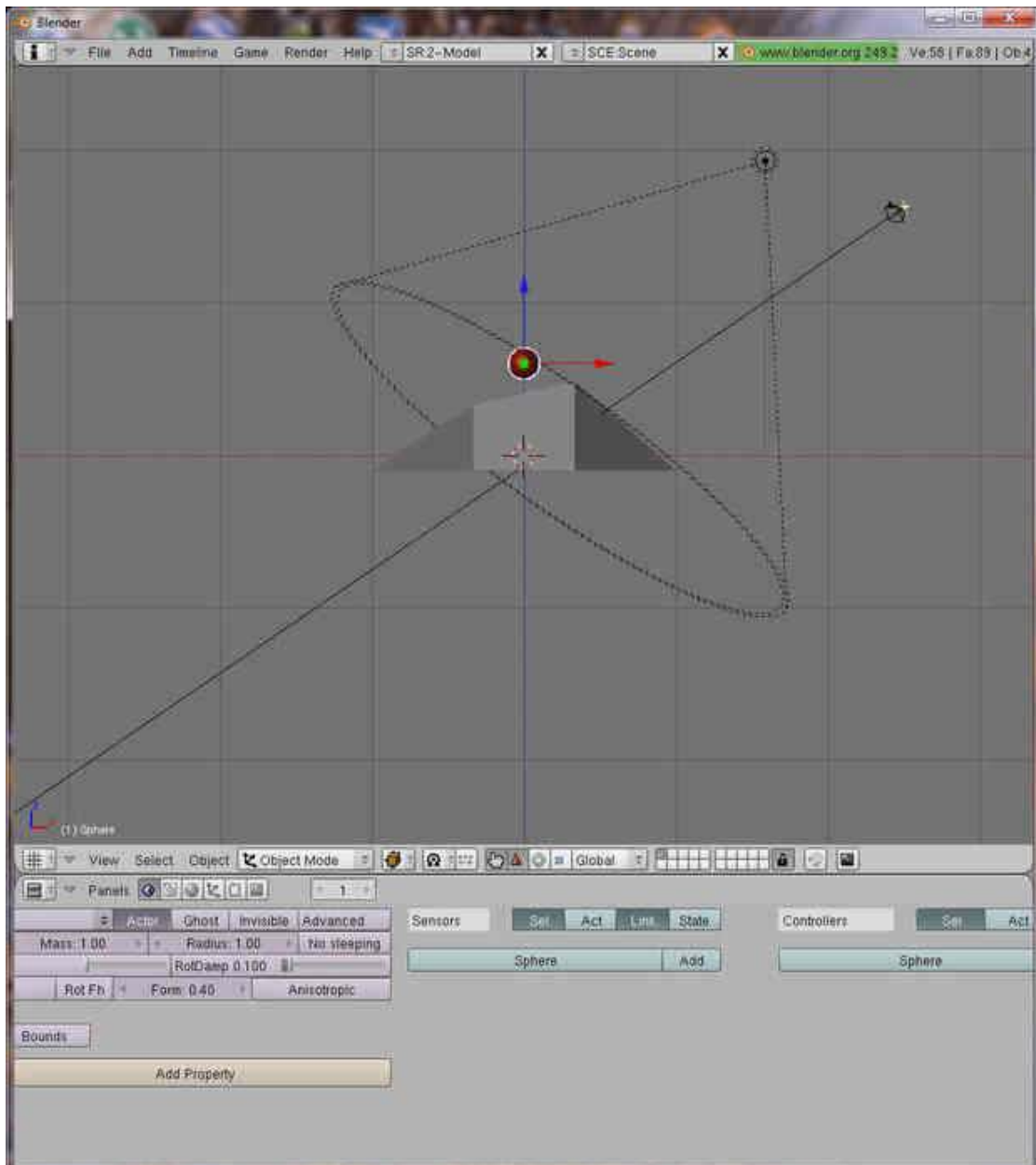


Figure 872 Rolling ball tutorial screen capture

110.4. Video capturing your game

When you press the PKEY or click *game, start play*, Blender will play it using the 3D view. Many rendering features are not shown in this 3D Window and it does not render the view in order to get good pictures with textures and lighting. You must capture your object that keep on changing(the actor) so that it can be animated.

View -> click View button , -> next step / -> alternative step RMB -> right mouse button IKEY -> press the I key on keyboard

Split window, *IPO Curve Editor View, Game, Record Game Physics to IPO, object mode, RMB(select actor), IKEY/select key frame, Loc(location of actor object only), PKEY(play game), ESCKEY(stop game)*

scene(F10), output, f:\animation\ball (your file name),stamp, time, date, draw stamp, format,(choose your output format)

anim, end(ending frame), 270, step, 10(for testing), ANIM/Render, Render Animation/CTRL F12

110.5. Conclusion

With the knowledge acquired in this tutorial, there are many things you could accomplish within the Blender Game Engine, although the majority of them would require more knowledge. So read on, and work your way through the multitudinous seas of tutorials (That is, two).

110.6. Extra Tutorials

Making a Basic Game: [Link¹](#), The State Actuator: [Link²](#), Blender Game Engine Mouse Follow: [Link³](#), Blender Bullet Physics: [Link⁴](#), Domino Game: [Link⁵](#), Rag Doll: [Link⁶](#)

1 http://www.youtube.com/watch?v=WGycLDs_dLg
2 <http://forum.nystic.com/viewtopic.php?f=9&t=4633>
3 <http://forum.nystic.com/viewtopic.php?f=9&t=4743>
4 <http://forum.nystic.com/viewtopic.php?f=9&t=4752>
5 <http://forum.nystic.com/viewtopic.php?f=9&t=4820>
6 <http://forum.nystic.com/viewtopic.php?f=9&t=4987>

111. Platformer: Creation and Controls

This tutorial will teach you the basics of the Blender Game Engine, and how to create a platformer game, which is a game where you overcome in-game obstacles by controlling the character's motions. (e.g. Super Mario)

111.1. Set Up

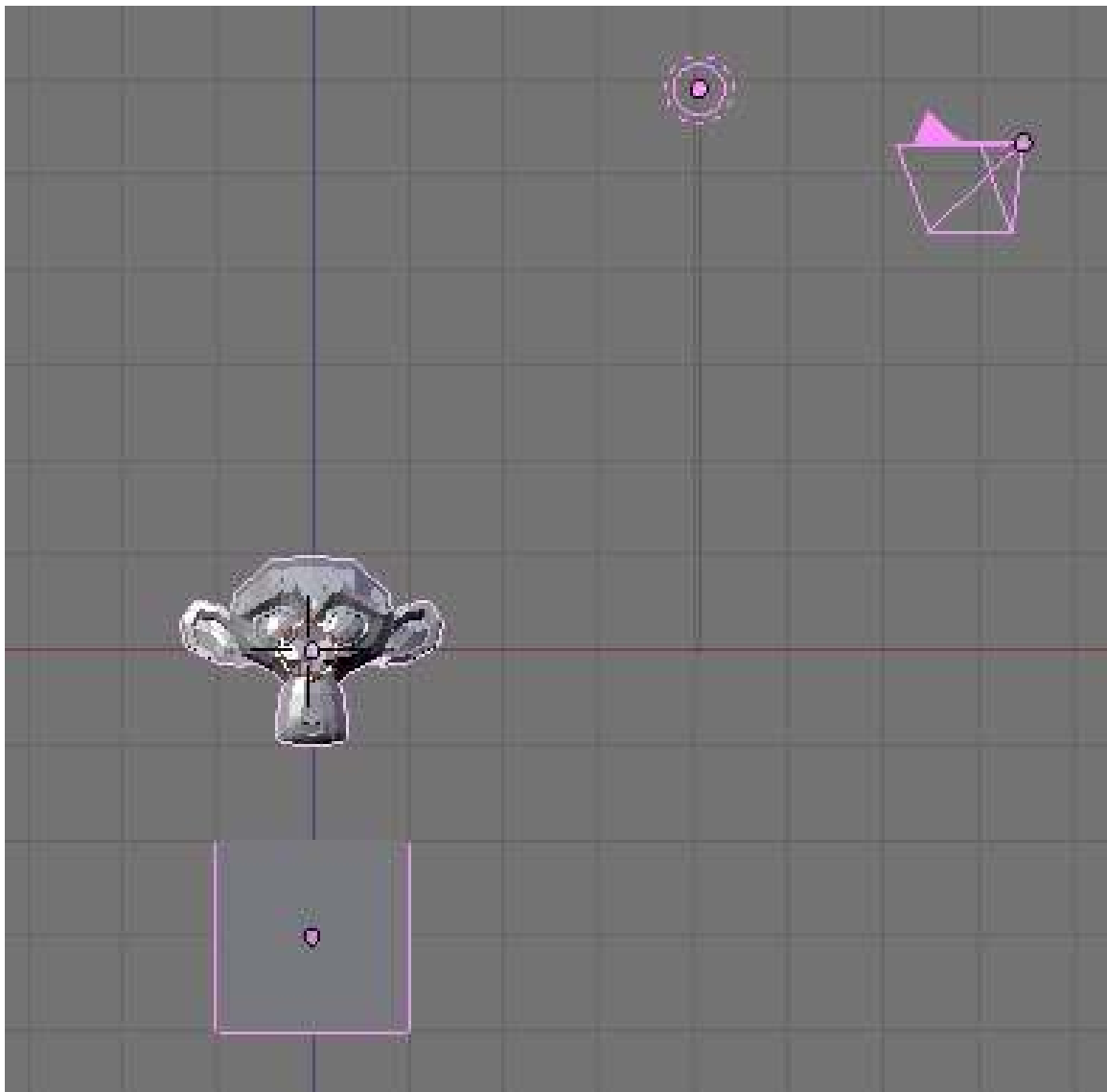


Figure 873 How it should look so far.

To start, switch to the frontal view **NUM1** and move the starting cube to -3.0 along the Z axis, or three spaces down. If your default settings do not have you starting with a cube, create one now. Next, hit **SPACE** → *Add* → *Mesh* → *Monkey* to create Suzanne the Monkey. Any mesh will do, but it's the easiest of the basic meshes to intuitively determine local direction from. Next, go to the logic window (with Suzanne still selected) and toggle on "Actor" and then "Dynamic". Hit **PKEY** to test. Suzanne should fall, then stop upon hitting the cube. Press **ESC** to exit.



Figure 874

111.2. Controls

Now we will add the controls. First, forward:

In Blender 2.59 these controls are available in the 'Logic Editor' window.

1. Create a Keyboard Sensor, AND Controller, and Motion Actuator. Connect them by dragging lines between the dots next to their names. (In 2.59 the controller is created automatically, if the other two are connected)
2. Select the empty box next to the word "Key" on the Keyboard Sensor and hit the up arrow key on your keyboard.
3. Set the dLoc on the Motion Actuator to 0.10 Z and toggle Local Transformation (may appear as simply an L next to the dLoc row).

Next, backward. Repeat the process used for forward, but set the dLoc to -0.10 Z and the key to the down arrow. You can repeat this for left and right if you so choose to have a linear control scheme. We will not be using such a scheme for this tutorial, but you should still be able to follow along just fine. Instead, the character will rotate left and right and move in the new forward and backward.

To rotate right:

1. As before, create a Keyboard Sensor, AND Controller, and Motion Actuator, then connect them.
2. Set the key to the right arrow and set the dRot to -0.10 Y.

Repeat for left, with the left arrow and 0.10 Y.

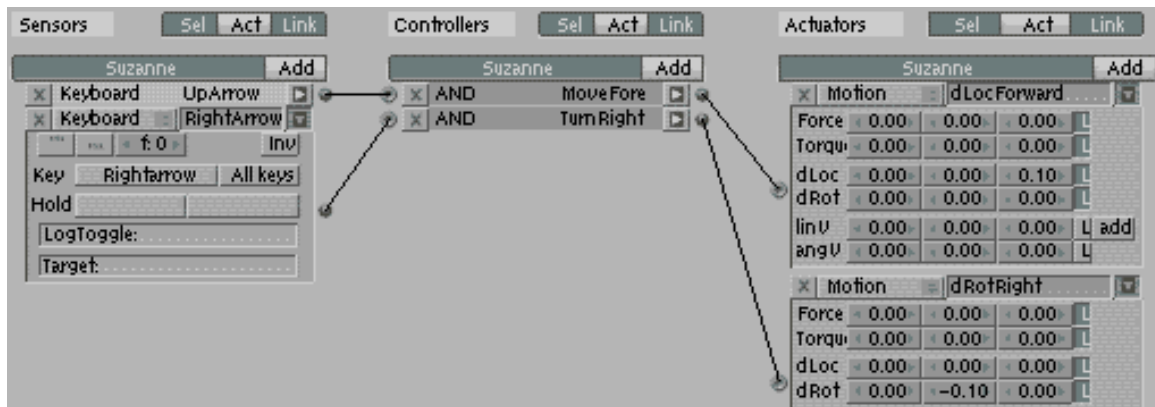


Figure 875

Finally, the ubiquitous, and some say defining, Platformer move: the jump. Repeat the previous processes, with the space key as the trigger and the linV of the motion effect as 7.50 Z (not local). This doesn't actually move the character in a certain direction, but sets its velocity. If you've done programming for games before, this will probably be familiar to you. Each frame, the engine adds the velocity of the object to its current position. The gravity portion of the engine subtracts from the upward velocity each time. When it reaches a negative, it's adding a negative number to its altitude, ergo subtracting from it, ergo causing it to fall. This is why using linV to move the character does not cause it to immediately relocate to the target position.

112. An aMAZEing game engine tutorial

This tutorial is intended as an intermediate introduction to the Blender game engine, in the form of a game, and is the sequel to *Platformer: Creation and Controls*¹. It will require a familiarity with the Blender UI, simple commands (such as **AKEY** to select) and basic modeling skills. The game we will create within Blender will have the following features:

1. a protagonist controlled by means of the WASD keys
2. a maze surface without walls
3. death on falling off the maze
4. multiple levels
5. dynamic obstacles
6. a goal within the maze that will transfer you to the next level

This tutorial was written for version 2.44 and has been tested for all later versions (as of April, 2008)

112.1. Maze Surface

Now we'll make the maze. When you are creating the path of the maze surface keep in mind that you need a start and a goal.

¹ Chapter 111 on page 937

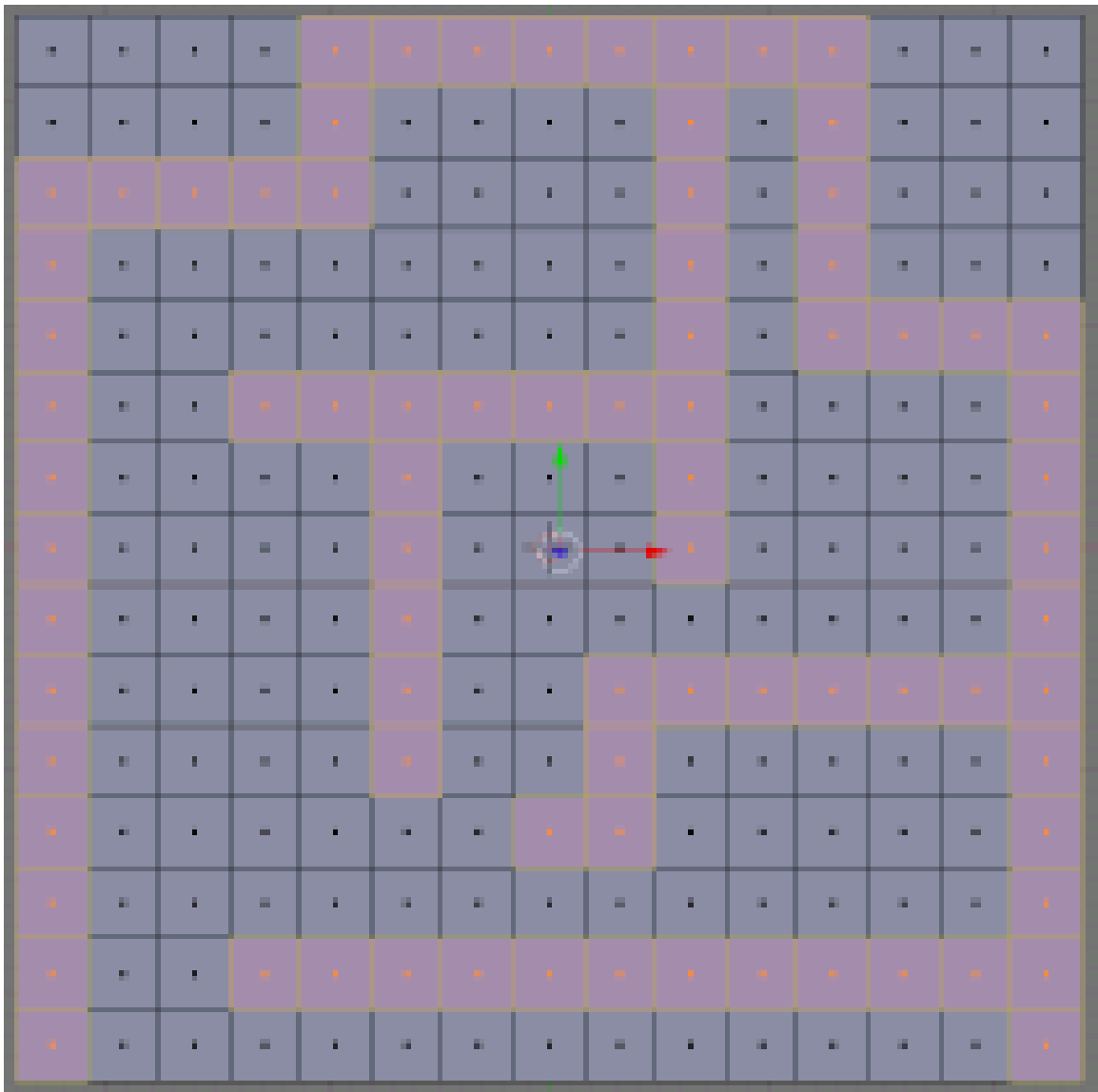


Figure 876 Selecting the faces

1. Clear the scene; Select all (**AKEY**), delete selected items (**XKEY** → **ENTER**)
2. Go into top view (**NUM7**)
3. Add a grid (**SPACE** → Add → Mesh → *Grid*) with X and Y resolution of 16
4. Scale grid by 24 times (**SKEY**, type 24, **ENTER**)
5. Enter edit mode with the grid selected (**TAB**)
6. Go into face select mode (**CTRL+TAB** → *Faces*)
7. Start selecting faces so as to make a two dimensional maze (use **BKEY** to draw selection boxes)
8. With the faces selected, duplicate those faces (**SHIFT+D**, **RMB**)
9. Move those faces away from the original grid (**GKEY** or the movement widget **CTR+ALT+GKEY**)
10. Delete original grid

- Center your maze using **GKEY** or by pressing the **Center** button in the **Mesh** panel of the Editing buttons (**F9**)

You should now have a two dimensional maze. In the game engine, we only see one side of every face, so this maze will not appear to be there when seen from below, and will be obviously two dimensional when seen from the surface of the maze itself. This is undesirable so we will make our maze 3 dimensional by extruding the maze surface down.

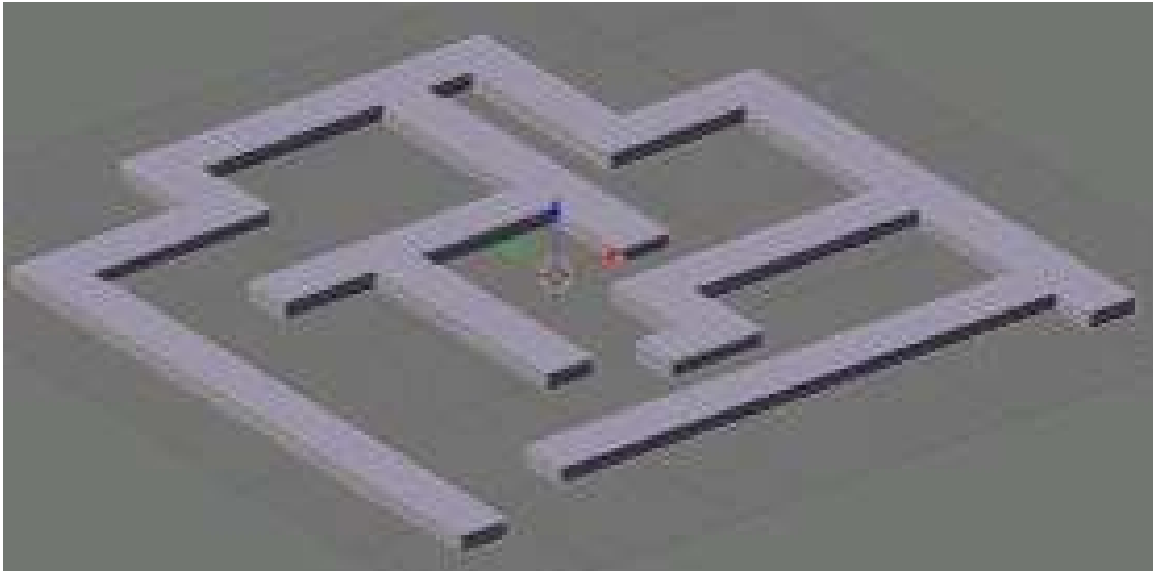


Figure 877 The extruded maze

- Go into side view (**NUM3**)
- Select all the faces (**AKEY**)
- Extrude down, -1 unit (**EKEY** → *Region*), hold **CTR** to constrain to 1 unit increments, or manually type in the value -1, like you scaled the grid in the beginning.
- LMB** to accept the operation, or **RMB** to cancel the move of the faces. If you cancel it, the faces will have extruded, but will simply not have moved. You can undo the operation (**CTRL+Z**), or delete the vertices/faces you have extruded (**WKEY** → *Remove Doubles*).

Your maze is now complete, only lacking someone trapped in it...

112.2. Character

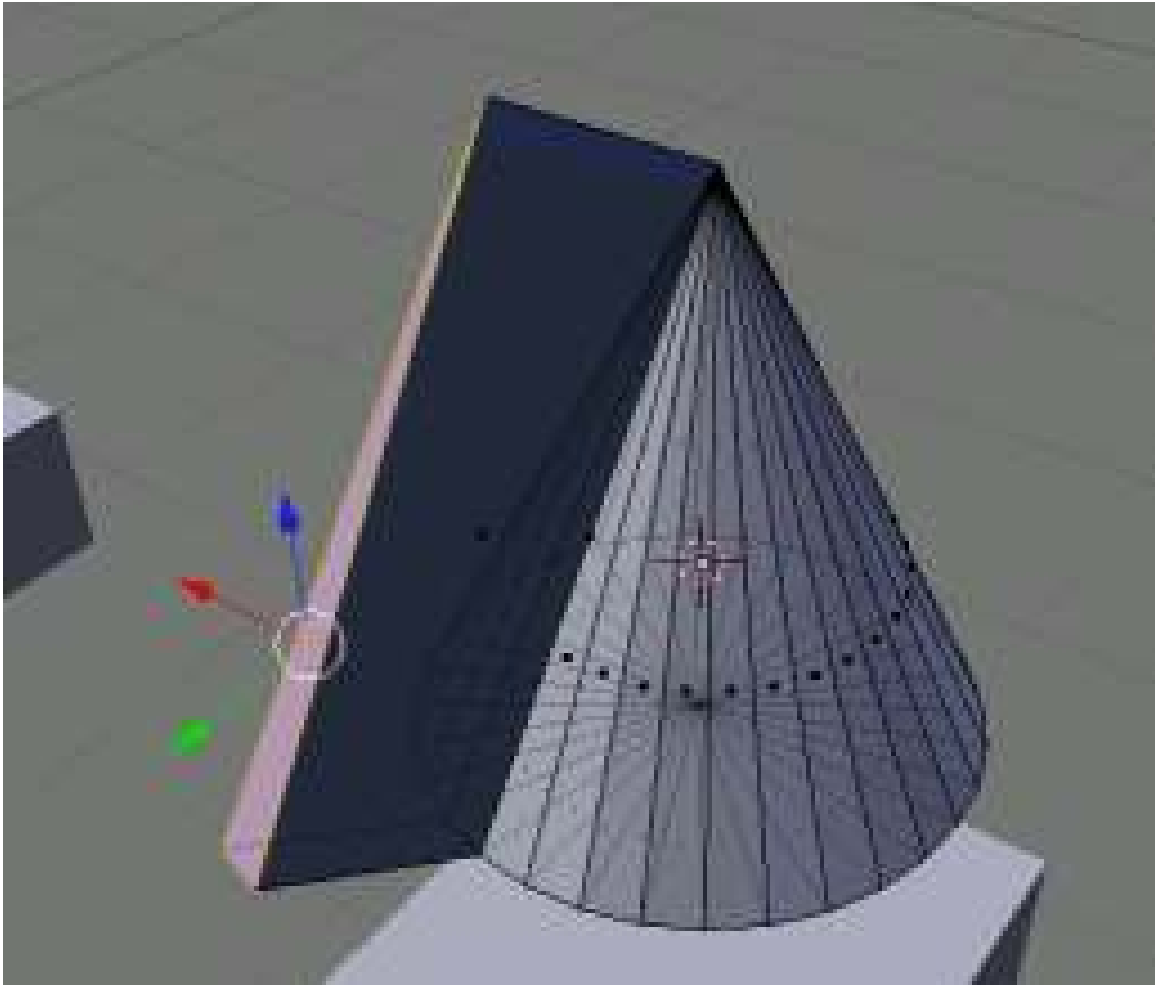


Figure 878 Extruding a nose

We will create a very basic character sufficient to illustrate the concepts meant to be conveyed in this tutorial. You can make a character as complex or basic as you desire, but it is important to always have an indicator of direction on your protagonist, otherwise the player may become confused. So, we will use a primitive with a 'nose' (i.e. a face or vertex extruded away from the object center in order to provide a point of rotational reference). I'll use a cone for this tutorial, simply because they don't roll very easily.

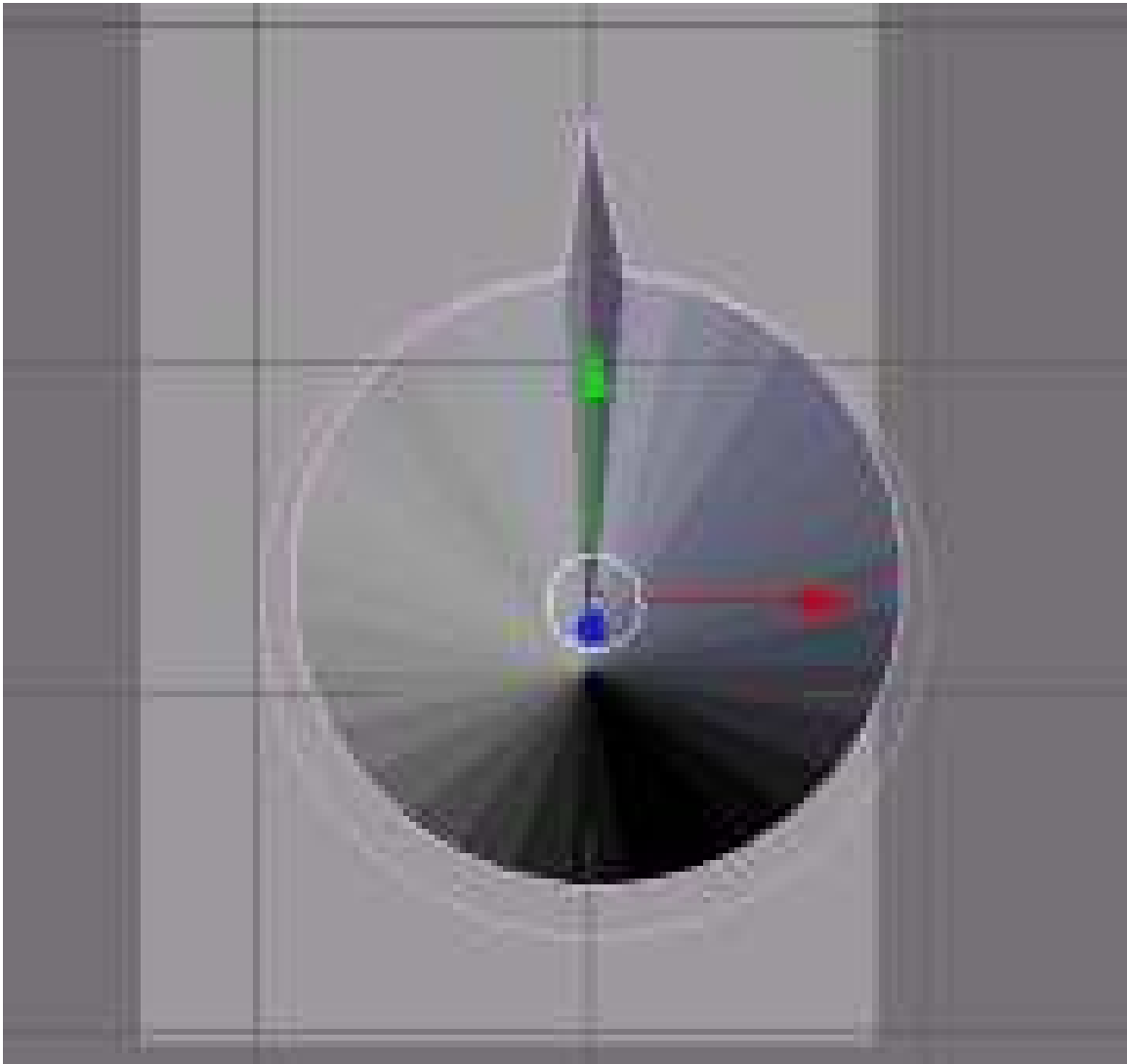


Figure 879 Rotate to the direction character will move

1. Enter top view (**NUM7**)
2. Add the cone (**SPACE** → *Mesh* → *Cone*, use default values)
3. Select cone and enter Edit mode (**TAB**)
4. Enter face select mode (**CTRL+TAB** → *Faces*)
5. Select a face on the side of the cone
6. Extrude it about 1/4 of a unit or less (**EKEY**)
7. Merge the extruded face to make the nose pointy (**WKEY** → *Merge* → *At Center*)
8. Enter object mode (**TAB**)
9. Rotate the cone so the nose is aligned with the positive Y axis (**RKEY**, align with the green arrow)

You've got your character now (Not exactly Pixar-like, but it will do for now). Now we just need to give it motivation.

Note: From points 8 to 9 it would seem that the cone should rotated so that the “nose” is aligned with the Y-axis in object mode, but after doing this I encountered the cone still moved in the original Y-axis direction (before rotation). To remedy this I did the alignment of the “nose” to the Y-axis in edit mode instead, this seems to have solved the problem.

112.3. Motion

We'll now make the character move, via the use of the WASD keys. In order to do this, game logic needs to be associated with the WASD keys. Copy the configuration seen in the screenshot, taking care to include the convex hull polytope collision in the top left as well as depressing the 'L' buttons, which make the force and torque local to that object rather than global. If the force was global, the protagonist would continue moving in a straight line, even when turning.

1. Go to the Logic buttons (The purple pac-man icon or **F4**)
2. Add four Sensors, Controllers and Actuators.
3. Maximize the logic window (**CTRL+UPKEY**)
4. Copy the configuration seen in the screenshot below.

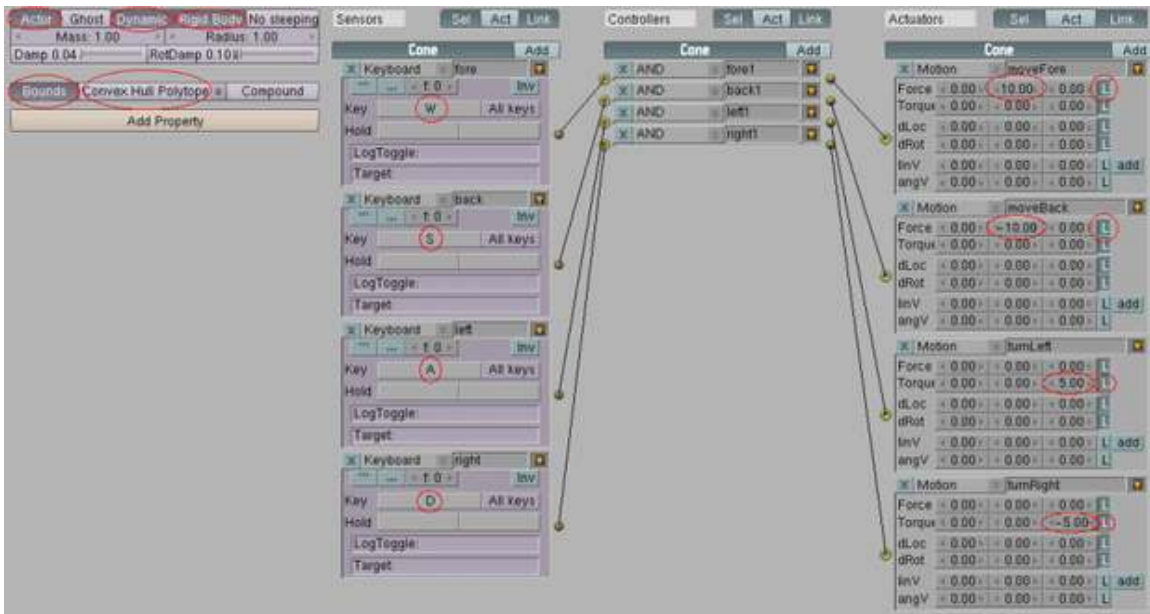


Figure 880

(Note: I made the simulation using Actor->Dynamic since there was no way to use both dynamic and rigid body in 2.49b. After some testing it appears to work)

dLoc/dRot vs Force/Torque

The character logic in this tutorial is very similar to that of the previous platformer logic, but it differs in that it uses force and torque rather than dLoc and dRot, respectively. The difference between these two is that force and torque move and rotate the object within the bullet physics engine, applying friction and collision to the process, whereas dLoc and dRot move and rotate the object without regard for any other 'actors' (objects factored in to the physics engine's calculations). Although dLoc and dRot would serve in most maze games, many other instances demand force and torque.

However, most of these instances occur in more complicated games with a multitude of interactions with dynamic objects. Later in this tutorial we will add dynamic objects to our maze, and the reasons for using force will become clear.

112.4. Testing

Now the time has come for the first test of our game.

1. Add a light source well above the maze (**SPACE** → *Lamp* → *Lamp*), align in front view (**NUM1**)
2. Press **NUM5** to enter perspective viewmode, which gives a realistic view, rather than a view in which objects stay the same size with distance. Be sure to switch back to orthographic view when you are editing (**NUM5**)
3. Enter textured mode (**ALT+ZKEY**) -- press **ZKEY** to switch back to solid view mode
4. Switch into top view (**NUM7**) or camera view once we install the camera (**NUM0**)
5. **PKEY** to play the game
6. **ESC** to escape testing the game

If you receive a "No camera" error, disregard it. We'll be adding a camera later.

If your character doesn't move at all make sure you drew the connecting 'wires' between the actuators in the logic panel, as well as setting the "Actor" "Dynamic" and "Rigid Body" buttons to True.

You should now have a nosy cone that speeds around the maze.



Figure 881

(**Answer:** If I understand your question right. If i used torque to turn and force to move forward and pressed forward then turned it started spinning out of control. My solution was to use rotation instead of torque.)

112.5. Falling

We will now implement the "death on falling off maze" feature. In order to accomplish this, we'll add something that allows us to check whether the cone has fallen off the maze or not. The simplest way to do this is with a 'skydome' object the entire maze is within, and although there are a number of more elegant ways to accomplish the goal, we will use the skydome method for its simplicity to replicate. From now on we'll need to test the maze with both layers one and two enabled (**SHIFT+2KEY**)

1. Add a material to your cone called coneMat (**F5**, press "Add New", Links and Pipeline, Link to Object, MA:coneMat).
2. Add a new cube
3. resize it to gargantuan proportions (**SKEY**, just large enough to encompass the maze, and all within it)
4. Move it to layer 2 (**MKEY**, **2KEY**, **Enter**)

Now that you have a massive cube, go to the logic panel and add one Sensor, Controller, Actuator set, and set them as seen in the screenshot.

1. Sensor: Touch, coneMat
2. Controller: And
3. Actuator: Scene, restart



Figure 882

Enable the actor button and press "ghost" this will let the dynamic obstacles, which are added later on, fall through and not get stuck there!

If you test you game now, you should see the game be restarted when you fall off the maze. As we discussed earlier, faces are one-sided in the game engine, so you should not be able to see the cube.

(Answer: The tutorial says to create the cube in layer 2, although the layer shouldn't matter. Make sure you test the game with both layers one **and** two selected. Also double check the logic for the 'skycube')

(Answer2: I had the same problem. I fixed this by replacing the cube with a grid, scaled and placed under the maze. The grid had the touch logic mentioned above.)

(Answer3: Same here. I thought I'd remove "coneMat" from the material reference, but that wouldn't do if the bricks fell down and touched the cube, it'd just restart the level... the solution: create a property in the cone, a bool value named "isCone" (=true), then in the cube add a sensor 'collision' that checks for the object's property 'isCone' and keep the rest as shown.)

(Answer4 (to restart when block hits cube):Do all as the pic shows you and RMB the cone and go to links and pipelines and rename the MA: to cone or whatever you want, then RMB the cube then in the Touch sensor then rename the MA: to whatever you called the other MA: . :D this is in 24.9b)

(Answer5: For blender 2.57a you need to place a plane or something under the maze. Then give it (touch -> and -> Scene(restart)) game logic. If it isn't working make sure that actor is selected under physics tab.)

(Question 2: How do I make the box translucent? Right now all I see, also when testing, is the skybox. Also, when I put the box underneath the maze, it works fine, but then none of the objects have shading, while when not showing layer 2 everything looks shaded. Can anybody give a solution? This is in 2.48a)

(*Answer:* To make something transparent- press **F5**; press the "Material Buttons" button; under the Links and Pipeline tab press Halo and ZTransp, unpress Shadbuf; under Material make A 0.000.)

(*Answer 2:* My solution was to press the Invisible button (Logic section, next to Ghost in 2.49))

I have put in a new way to die. Add a static Plane, give it a property called Killer and place it just below the level. Add an ALWAYS sensor and connect it to a MOTION actuator that has 0.01 as the top last value. This way it will sloooowly come up. Now, select the player and add a COLLISION sensor. For the property, add Killer. Connect it to a GAME actuator, select Start New Game. You've just added some tension to your game! If you run out of time, you will touch the floor and restart! If you need more time, just put another 0 in 0.01 so it will look like 0.001 Sorry if you can't understand it very well, this is the first time I edited a page on Wikipedia + I don't speak English very well. Have fun! :)

112.6. Camera

SPACE → *Camera* near the protagonist cone. There are two easy ways to do the camera: logic camera and child camera. In this case you might be better off with the logic camera, but it really is a matter of personal preference.

- **Logic Camera:** Add to the camera's logic panel
 1. Sensor: Always
 2. Controller: And
 3. Actuator: Camera, to object "Cone" (or whatever your protagonist is called--look under the object panel when you have the protagonist selected) height 5, min 5, max 10.
- **Child Camera:** Select the camera, then select the protagonist (order is important) then **CTRL+PKEY** to parent the cone to the camera. Align the camera in a view that you like, and test the game.
- **First Person Camera:** Well, this one doesn't really count as another method, but if you're willing to redo your motion to just move the camera instead of the character, (just do everything you did for the character for the camera) you can run around in the first person. Or you can just put the camera in the cone because one of the sides always are invisible (look in texture mode) and parent it to the cone. **REMEMBER:** To access the ingame camera during gameplay, before you start, hit ctrl 0 or you can go to (view > camera)

(*Answer:* Ctrl+0 assigns the selected object as a camera. To solve your problem select your camera and press ctrl+0, to enter camera view press 0)

112.7. Beautification

Blender is a great 3D modeling program, and with such a vast number of tools at your disposal, you should be able to make your game look better than this:

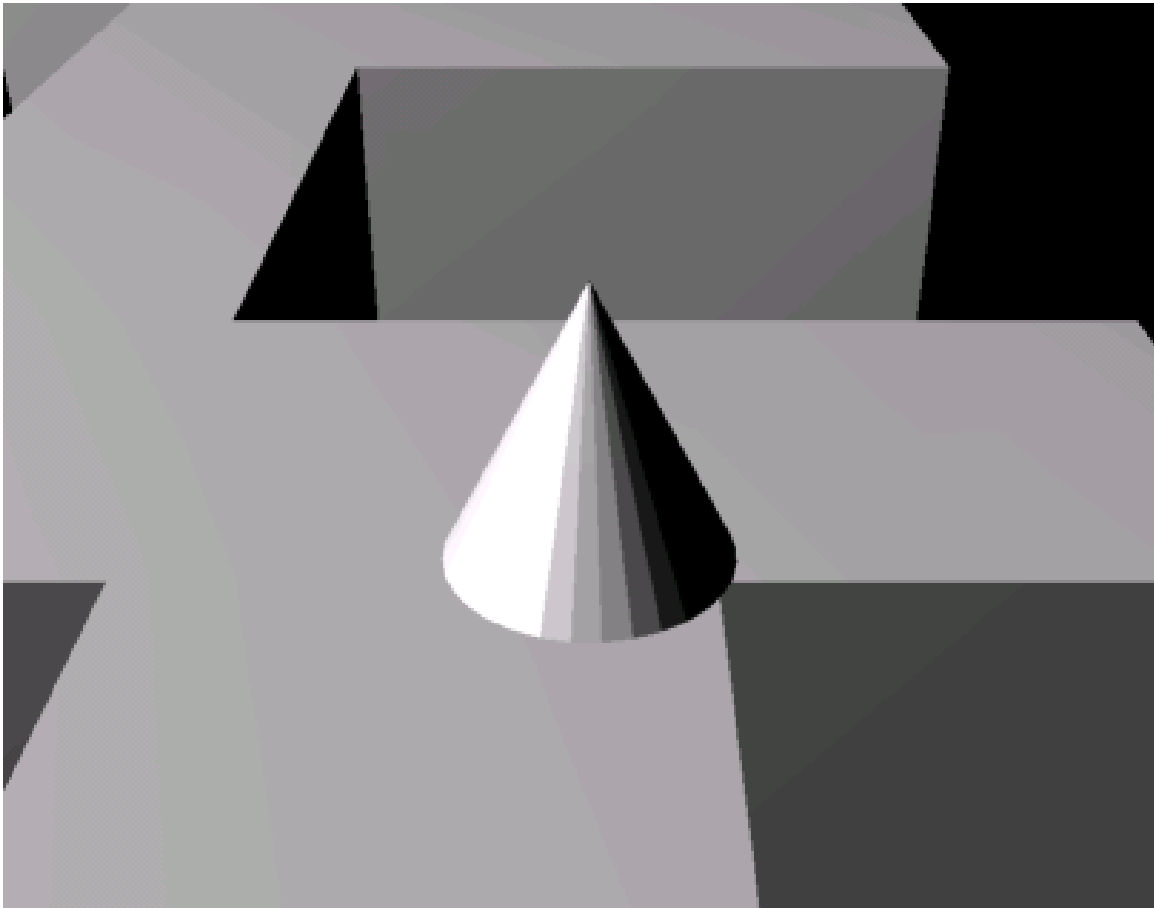


Figure 883

We'll add one or two textures to each of our objects.



Figure 884 Press the 'new' button then the 'assign' button to assign a material to a number of faces

1. Select the object you want to texture
2. Go into edit mode if you want your object to have more than one color
3. Select all the faces you are going to assign to a color (**CTRL+TAB** → *Faces*)
4. Go to the edit panel (**F9**) on the buttons window
5. Near the left there should be a panel that says "Link and Materials". Click on the buttons highlighted in the screenshot
6. Adjust the color of that material by going to the material buttons and changing the "Col" (color) value to something other than white

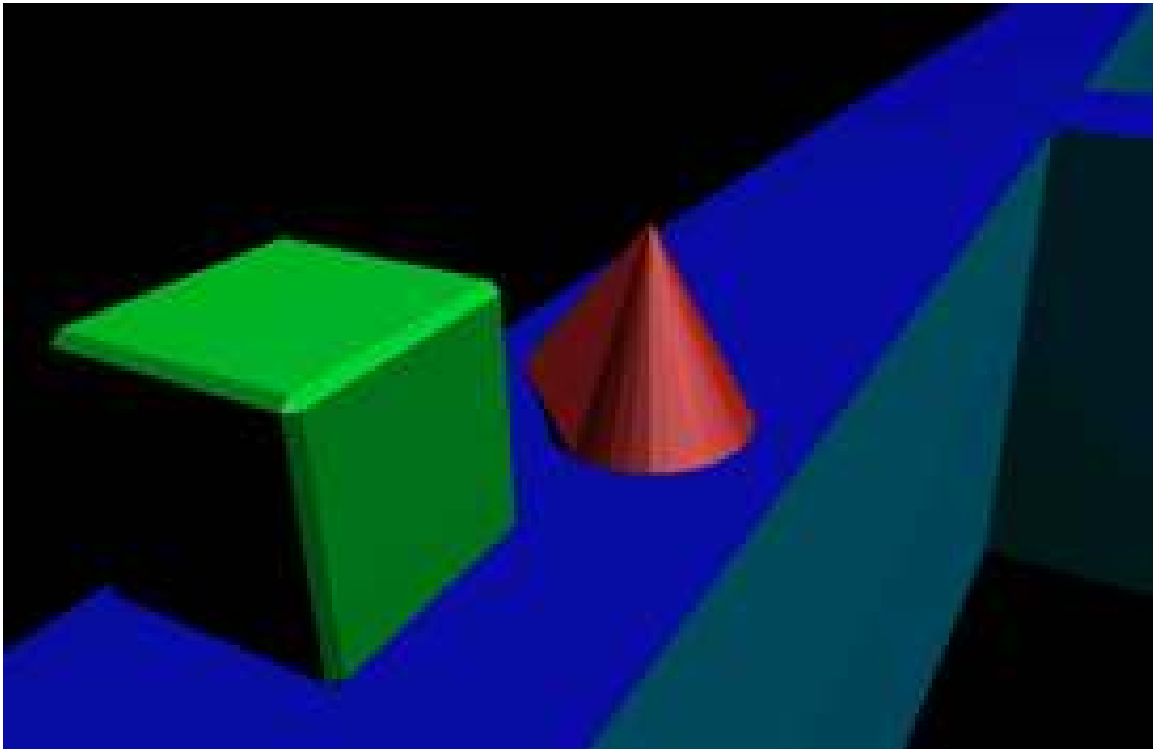


Figure 885 Maze game with materials

It looks better now, but when the camera is nearly horizontal, you can see where the camera stops rendering the maze. You can fix this by selecting the camera and, in its edit buttons, changing its clipping range... but there is a better way. Mist.

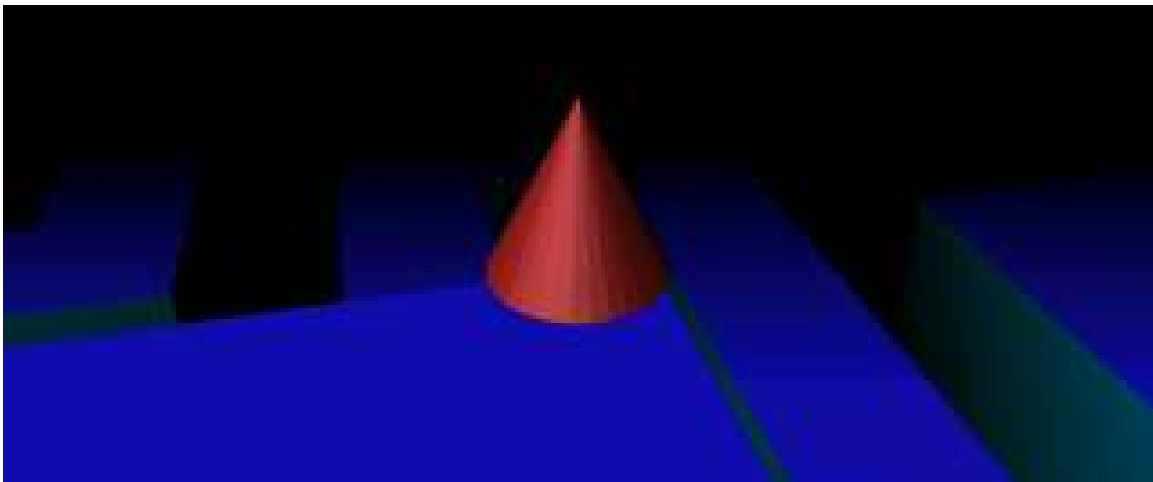


Figure 886 With mist and raised sections

Mist obscures everything a certain distance away, is the same color as the world texture, and can be handy. We'll add some mist to our game so the player can't see too far ahead of him, thus making the game too easy.

We'll make the mist quite close, although you can vary the distance. Go to the Material button, and to the World sub-buttons, then copy the settings in the screenshot. **When you test this out, make sure you are in texture mode to see the mist.**



Figure 887

(Answer: Change Start to a smaller value like 15.)

(Question: Is the screenshot correct, or are my sizes different. I found that a mist distance of 10 was barely visible, yet a distance of 1 had a much better effect.)

(Question: I have been able to create the mist, and can see it when I start the game, but the background colour is still grey, so the maze is obscured but everywh

112.8. Levels

Now all that remains is to create the next level with some obstacles.

1. Add a cube at the end of the maze, just above the floor level so the cone can touch it (**SPACE** → Add → Mesh → Cube)
2. Press the Scene dropdown menu and select "ADD NEW" → Full copy
3. You can now go back to the Maze Surface section to make a new maze mesh
4. Go back to level one via the Scene menu
5. Select the End of level cube
6. Change its logic panel to resemble this:

- Sensor: Touch, coneMat
- Controller: And
- Actuator: Scene, Set Scene, Scene.001 (or whatever you named your level 2)

(noob: In my level 2 my cone hits the cube and nothing happens!) Answer: Make sure that the Sensors, Controllers and Actuators tabs under the Logic panel (F4) are connected with wires.

(noob: When I went to level two I can't see anything!

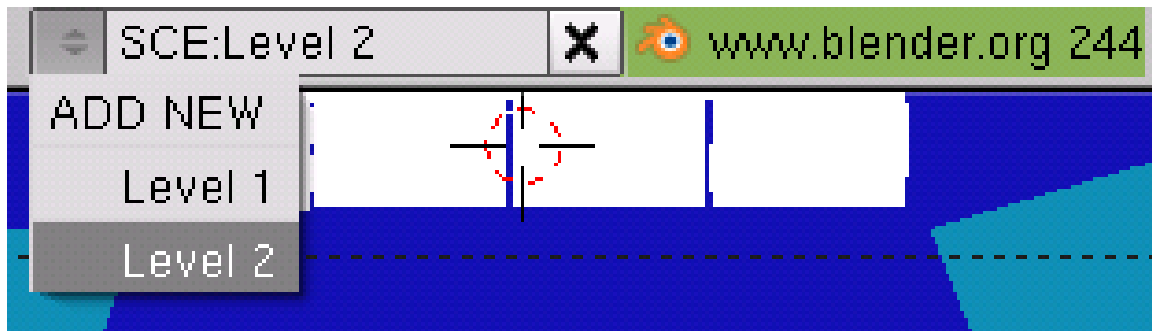


Figure 888 The scene button is on the top panel

To make it so multiple objects have to be touched (or acquired by the character, as it may be), give each object the logic settings described above and change the actuator type to Message with the subject "goalget". If you'd like the object to disappear after it is collected, add an Edit Object actuator to the object of the type "End Object", and link it to the existing AND controller.

Next, select the massive cube that restarts the level upon contact with the character. If you chose not to include one such cube, create an empty at the start of the level (so you can easily find it again later). Give the cube or empty the following logical operators:

- Sensor: Message, goalget
- Controller: And
- Actuator: Property, Add, wincon, -1
- Sensor: Property, Equal, wincon, 0
- Controller: And
- Actuator: Scene, Set Scene, Scene.001 (or whatever you named your level 2)

Then click "Add Property" and give it the Type Int, the Name wincon, and a value equal to the number of objects needed to be collected to complete the level (if all of them, make it equal to the number contained in the level). This can be expanded to include other multiple win conditions besides the collection of an item by making the completion of each condition subtract from "wincon".

You now have a multilevel maze game. Feel free to elaborate on the gameplay mechanics and models so they look better.

Noob note: I typed in where i thought everything should go but it's not doing anything. I'm not sure of what I'm looking for while in playmode.

112.9. Dynamic obstacles

This part of the tutorial will assume increased skills and knowledge of the Blender interface and basic commands. As such, this section will be written less specifically, and the shortcuts for all basic commands will not be covered here. From this point on, this tutorial is of an intermediate level. If you cannot follow this part of the tutorial, you can learn the basic functions used below in an earlier tutorial².

² http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Learn_to_Model

We will now add some dynamic obstacles. In this case, they will constitute a 'brick' wall that can be broken through by your character.

1. Add a cube
2. Scale it (in edit mode) to the dimensions 1,0.5,0.5
3. Bevel it (WKEY) recursion 1, size 0.03 (press space to enter value manually)
4. In logic panel, set it to Actor, Dynamic, Rigid body, Mass 0.5
5. Go right below that to click Bounds and leave it on "Box". That makes Blender realize that what you added is a box, and should act like it (instead of rolling, it will slide now)
6. Move it to an intersection in your maze (GKEY) and duplicate it until you have a brick wall (ALT DKEY, LMB)



Figure 889

You now have a barrier that your character can break through.

By using this

112.10. Conclusion

You have now learned enough of the Blender Game Engine (BGE) to create your own game. If you wish to go very far into Blender games, I recommend you learn Python, as trying to make a full game within the graphical interface for the BGE is like trying to dig a grave with the blunt half of a toothpick. Blindfolded. And the toothpick is glued to your forehead. However, you are now well on your way to becoming a game-maker!

If you have any questions, concerns, or 'noob notes', please post them in the discussion page for this tutorial, and not in the tutorial itself.

113. Platformer: Physics Fixes

This tutorial is intended as an improvement on its prequel, Platformer: Creation and Controls¹, and will require files created through that tutorial. It is also recommended that you read An aMAZEing game engine tutorial² as well before beginning this tutorial. It will require a familiarity with the Blender UI, simple commands (such as AKEY to select) and basic modeling skills. This tutorial will introduce the following features and improvements:

1. only letting the character jump while touching the ground
2. stop the character from bouncing greatly upon hitting an object
3. where to adjust material friction

This tutorial was written for version 2.45.

113.1. Creating Your Hit Test Object

Copy the cube with **SHIFT-D**. Go into edit mode, select the bottom four vertices of the cube and delete them. Go back to object mode. Go to the editing panel in the buttons window. In the Link and Materials section, change the ME: value to "Square" and the OB: value to "FloorHit". In the Mesh section, there will be three buttons with the word "center" in them; Center, Center New, and Center Cursor. Press Center New. This will change the origin of the "Square" mesh to the center of all of its vertices, instead of the center of the cube you made it from. Scale it to 0.99% of the original and move it up 0.01 along Z. Now go to the shading panel and make sure you're in the Material buttons subpanel. Find Links and Pipeline, and make sure "ME is selected. Click the X next to the material selection (under the words "Link to Object"). Then select "OB" and press "Add New" and name this material "FloorHit" (again, no quotes). Make it green colored, so you can easily find it while editing, but turn on "Shadeless" and "No Mist" in the Material section and "OnlyCast" in the Links and Pipeline - Render Pipeline section and turn off "Radio", "Traceable", and "Shad(ow)buffer" in the Render Pipeline and "Shadow" in the Shaders section. This will make it completely invisible at runtime and take up little resources.

(question: it wont make it invisible it just flashes rapidly)

113.1.1. Making the sensor require floor contact

Now select Suzanne and go to the logic panel. Add a touch sensor called "jumpcol" (for 'jump collision') and connect it to the same AND controller as the jump keyboard sensor. Set the f to 10 and the MA: to FloorHit. This will make it so your character can only jump while it's touching something

1 Chapter 111 on page 937

2 Chapter 112 on page 941

with the FloorHit material. By that same token you can link this to *all* of your movement-related AND controllers so that the player can't adjust it's movements in mid-air. This is a nice physics touch but for most platformer games, where mid-air dexterity is almost essential, it doesn't work. You also have to use Force and not dLoc, otherwise your character won't be able to move at all while jumping, useless your character only needs to hit things with his/her head, and not jump over gaps. Note that this still does not keep the character from jumping while touching the hit test object at the sides, so they could still jump if they were touching the side of the collision surface, but at least they can't jump in mid-air or while touching the bottom of the ground. I have yet to find a solution to this problem not involving Python.

113.2. Excessive Bouncing

Another problem with the original model, as you might have noticed, is that when you run into one of the ground cubes you bounce back a great deal. To fix this, turn to the materials panel with one of your green hit test squares selected. Under the color selectors there should be three buttons that say "RGB", "HSV", and "DYN". Select DYN, and turn the restitution most or all the way up. As you can see, you can also find the friction property from here, if you want your ground to be more or less frictional (like mud or ice).

113.3. Final Notes

Always copy your floors, their respective hit test objects, and other generic objects with **ALT-D** and not **SHIFT-D**. This keeps you from duplicating things that remain the same for every one, like the material and mesh, which quickly drain resources if not recycled. Since there's not much further you can take the Platformer without learning Python, I suggest you start looking up information on the Blender API. Some essential information is to be found in the procedural object creation³ tutorial. A Blend file of the tutorial's finished product is coming soon.

Note: A (possibly) simpler method for jump limitation can be found on the discussion page for this tutorial.

3 Chapter 127 on page 1033

114. Making exe

This tutorial will show you how to make an executable for your game made in Blender.

Note: The methods listed in the "Windows" and "GNU/Linux or Mac OS X" sections work only for the operating system you are on when you create the file. To make it cross platform, use the "BlenderPlayer" method.

114.1. Windows

First create a folder that will hold all your game information. Name it something meaningful, like "Yo Frankie!".

The folder must contain four files that you can copy from your blender installation directory.(under windows it should be "C:\Program Files\Blender Foundation\Blender") They are:

1. SDL.dll
2. python24.dll (*Or other relevant python file; python25.dll, python26.dll etc.*)
3. pthreadVC2.dll
4. zlib.dll

(sometimes you will need the following also...

1. avformat-51.dll
2. avutil-49.dll
3. avcodec-51.dll

They are in the same folder.)

Not quite a Noob Note: If you are using the latest version of Blender you WILL need the following files)

I am running windows xp media edition and blender 2.46, here is a full list of files i needed:

1. avcodec-51.dll
2. avformat-52.dll
3. avutil-49.dll
4. libfaac-0.dll
5. libfaad-0.dll
6. libmp3lame-0.dll
7. libx264-59.dll
8. pthreadVC2.dll
9. python25.dll
10. SDL.dll
11. swscale-0.dll

12. vcomp90.dll
13. xvidcore.dll
14. zlib.dll

On Blender 2.48a (Windows XP Media Center 2002, SP3), these .manifest files are also required:

1. blender.exe.manifest
2. blenderplayer.exe.manifest
3. Microsoft.VC90.CRT.manifest
4. Microsoft.VC90.OpenMP.manifest

If you use *random* Python module in your game, you will have to add one more file to your game directory. For Blender 2.49 it would be python26.zip, which you can find in Blender main directory (where blender.exe is). Otherwise there may be some errors during executing the game in system without installed Python.

Since Blender 2.56 you will need to enable Save As Runtime; first open blender with the game that you have created and open the file menu. Click on User Preferences then select Add Ons then Game Engine, check the box Game Engine Save As Runtime and return to the file menu. Save As Runtime will appear as an option in the export menu then save to the new folder that you have created and rename the file yourgamename.exe,

and then you can run the game!

If you encounter errors during the "Save As Runtime" process (I'm on Windows 7) you can right click on the Blender icon and select "Run as administrator". So you will have your fresh .exe and you can spread it with a pice of your favourite peperoni pizza.

114.1.1. Making A Screensaver (Windows Only)

1. First, save your runtime (that's the .exe addressed above.) via File->Save Game as Runtime. ("Save Runtime" in older versions)
2. Now, go to your .exe game and rename it .scr - for instance, if your game was NotMyGame.exe, rename it to NotMyGame.scr. You can now right click to Install it, and then use it as your regular screensaver by applying it as you would any other screensaver (right click on desktop, properties... You know the drill.)

Just a helpful idea; when distributing my games, I often include a real awesome screen saver so that when Person A walks into Person B's room and sees that amazing thing going, they decide that they want my game. Just a note, though; screensavers are not games, and so they should not accept input. At most, they should be videos showing what your game does. If you just rename your regular game as a .scr, it will be remarkably boring, because your game needs input and screensavers do not.

114.1.2. Intel screensaver bug

There is a caveat with Intel Integrated Graphics drivers, found in many laptops though. The graphics driver shuts off OpenGL acceleration for screensavers, for some obscure reason. The way to work around this is to rename the .scr extension to a .sCr as the driver's algorithm is dependent on case-sensitive characters. If you experience very low framerates in your screensaver, you should attempt this fix, It is tested and reported to work.

114.2. GNU/Linux or Mac OS X

Open blender with the game that you have created and open the file menu. Click on **Save Game as Runtime** and then save to the new folder that you have created and rename the file yourgameName... and then you can run the game!

114.3. BlenderPlayer

The methods shown above only create an executable for your operating system. Well, BlenderPlayer can fix that.

1. Make a new folder to store all your game data.
2. Then save your .blend file into the directory.
3. You can skip this step and the next step if you do not want a Windows version. For the Windows users, copy blenderplayer.exe to the new folder from a Windows copy of Blender. Then copy all your DLL files for Blender as mentioned for Windows to the folder.
4. Next you have to make an MS-DOS batch file (for UNIX users, this is the shell script equivalent). In a simple text editor, in CR-LF mode if available (Notepad is always in this mode, and NOT a word processor!), copy and paste this text:

```
blenderplayer.exe yourgameName.blend
```

 Save it as YourGameName-Windows.bat in your game folder.
5. You can skip this step if you do not want a UNIX (basically Mac OS X and GNU/Linux) port. For GNU/Linux (at least), make a shell script. (A shell script is the UNIX term for a batch file.) In a simple text editor, in LF mode (unfortunately Notepad can't be used), copy and paste this text:

```
#!/bin/bash
./blenderplayer yourgameName.blend
```

 Save the file as YourGameName-UNIX.sh in your game folder. You will need to have BlenderPlayer in the same directory as the .blend game file.
6. Write a readme for your program. This is again best done with a simple text editor like Notepad or gedit, but it does not matter which mode it is in. You should include the name of the game, a description, perhaps a walkthrough or hints, and if you made a *NIX port, mention that it requires BlenderPlayer, available with Blender.

114.4. Legal Issues

Blender and the BlenderPlayer fall under the GNU General Public License. Blend files are copyrighted to their respected owners and do not fall under the GPL as long as they are not packed in the BlenderPlayer. If a user does not want the blend files to fall under the GPL, it is recommended not to use the "Save Game as Runtime" feature. To put it another way, the user must keep the blend files and the BlenderPlayer in separate files. More information can be found at

<http://www.blender.org/education-help/faq/gpl-for-artists/>.¹ (Blender 2.4 now has a run game from file actuator.)

114.5. Proprietary Blend Files

Blender has no built-in functionality to "lock" or protect user generated content. As a result, anyone who has Blender can open and/or modify blend files. However, it's still possible to lock blend files. Common methods involve encrypting the blend file and then temporary decrypting it at runtime. This can be accomplished by using python scripts or by using external 3rd party applications (which are feasible under the GPL v2).

¹ <http://www.blender.org/education-help/faq/gpl-for-artists/>.

115. Build a skybox

One way to add a realistic feeling to your 3d environment in a game engine is to create a skybox. A skybox is a large cube which has on its inside a projection of a 360° environment. When the player (camera) is inside this environment, the scene is rendered with the illusion of being inside a gigantic world. This is a similar effect to Quicktime VR (see <http://fullscreenqvr.com>¹ for examples). And, by setting up the skybox as a simple cube shape, you place the least amount of strain on the graphics engine. It's a great advantage for your game with very little overhead.

This tutorial will show you how to create skyboxes relatively easily from panoramic photos. My favorite part is, you can do it easily using free tools such as Blender and the Gimp.

Using the Gimp to manipulate images is not really in the scope of this tutorial... check out some other page on using that software. You should have an understanding of how to edit images and apply alpha channels. (You could also use the Gimp to apply a polar coordinate texture to your rectangular image in order to create a fisheye image. Hint: it's not the sphereize filter.)

115.1. Gather your graphics

You can take panoramic images yourself using a regular digital camera and a tripod. A quick way to accomplish this is to draw marks on your tripod base at every 30 degrees (think of the hours on a clock face). Make a single mark on the swivel of your tripod to allow you to line up your shots -- twelve shots at 30 degrees each. Then, using a program such as the Gimp² or the incredibly cool Autostitch³ to merge the photos into one big panorama.

Or, if you're lazy like me, you can just grab photos online to use as templates to create original images. There are also many places you can download non-copyrighted photos for free as well. One resource for cloudy sky textures, as well as panoramic photography instructions, is Philippe Hurbain's site Philo's Home Page⁴. This tutorial will use a fisheye sky photo from his copyright-free Panoramic Skies images collection.

You'll also probably want a photo for your ground, unless you prefer to use real models such as buildings in your skybox. This earlier chapter⁵ on creating landscapes can be incorporated into setting up your skybox. However, this tutorial will use the sky photo for the top half of our world, and a panoramic landscape with an alpha channel for the bottom half. I've created a ground image using copyright-free textures obtained from Accustudio⁶.

1 <http://fullscreenqvr.com>

2 <http://www.gimp.org>

3 <http://www.autostitch.net>

4 <http://www.philohome.com>

5 http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Mountains_Out_Of_Molehills

6 <http://www.accustudio.com>

Here are the images I'll be using (you'll want to use images with higher resolution): Note that the sky has trees, etc.



Figure 890



Figure 891

Note: I've outlined the horizon of the ground texture with an alpha channel which will allow me to place the ground mesh right against the sky mesh with a very natural feel.

115.2. Create a dome for the sky

Open a new file in Blender. Your default new file will probably be a two-unit cube in the center of the screen, with a single light source and a camera. You can delete the light source because we won't be needing it. Leave the cube, because that is what will become our skybox.

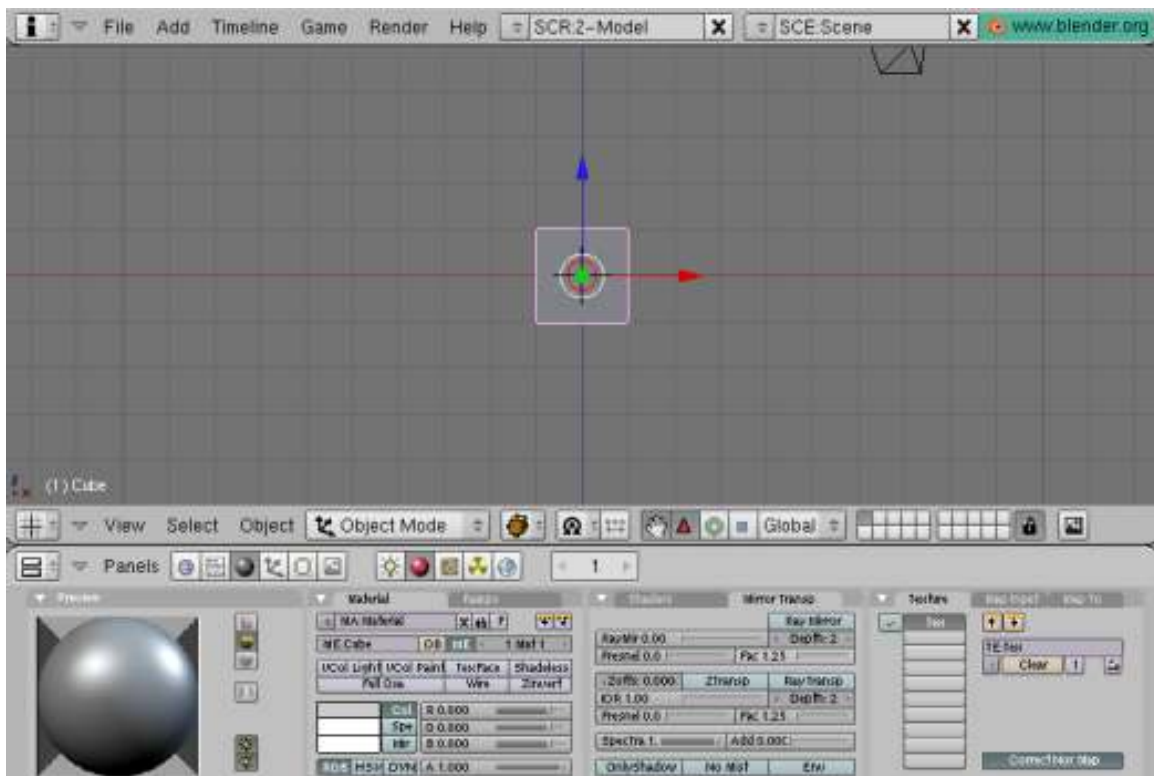


Figure 892 skybox tutorial 1

The cube will be the center of our environment, so use Object->Snap->Cursor To Selection if your cursor is not centered. Then, from the top view [KEYPAD-7], Use [KEY-SPACEBAR] to insert a new mesh; make it a UV sphere. I find a 32-segment, 32-ring sphere to be sufficient. We create the sphere from the top view because that is the projection from which we want to add the sky texture.

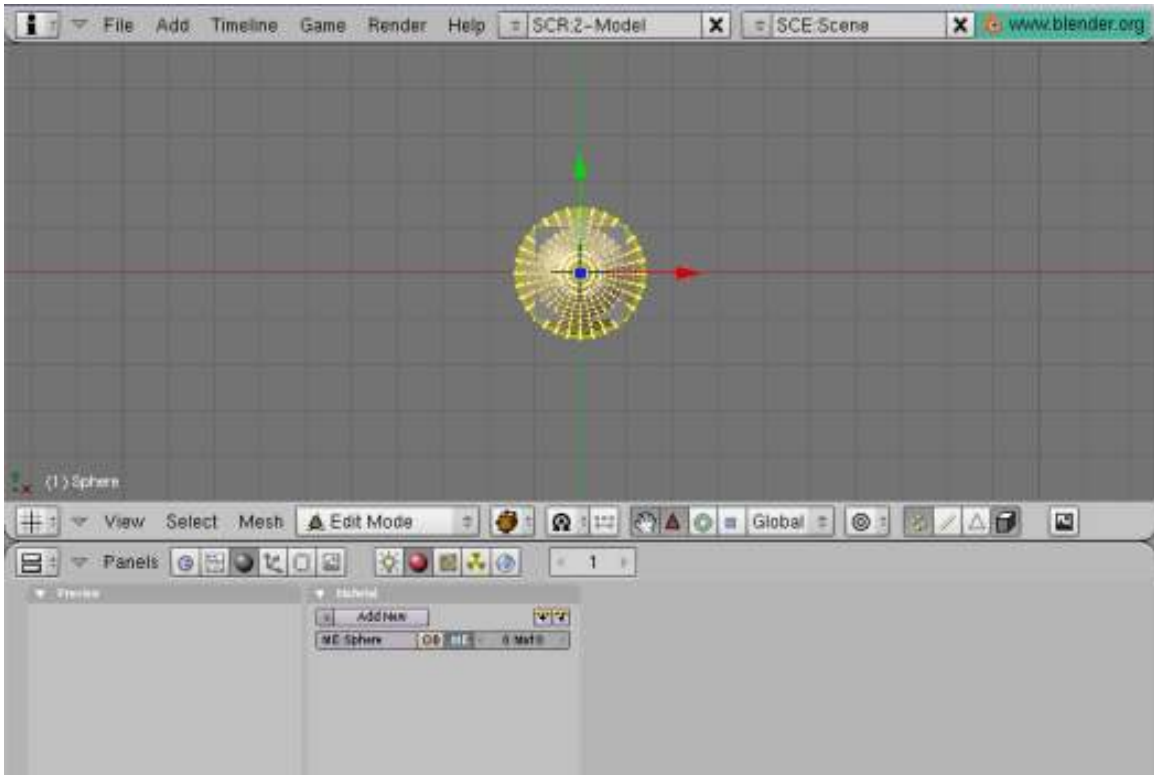


Figure 893 skybox tutorial 2

Scale up the sphere so it resembles a large "arena" in comparison to your cube, and select and delete the lower half of the vertices, using the front view [KEYPAD-1] and [KEY-B] to create a bounding box. It helps if "Select Visible" is turned off so you can select all of the vertices in one go.

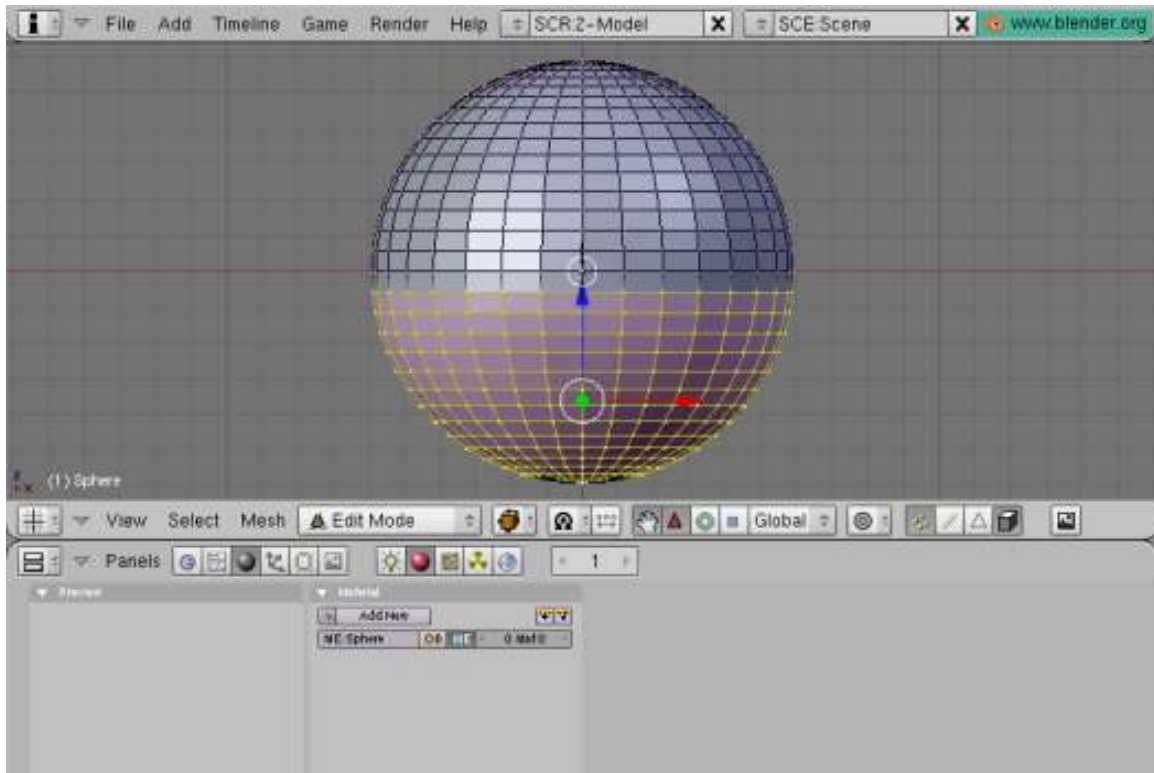


Figure 894 skybox tutorial 3

Turn on proportional editing with [KEY-O], then select the bottom row of vertices and scale them up with [KEY-S] so that the bottom of the sphere gets a bell shape. Because the projection of the sky texture will be from the Y-axis (ceiling) we need the bottom faces of the sphere to be at an angle, to catch the texture. (Faces perpendicular to the projection will look like smears.) Alter the influence of proportional editing with [KEY-PAGEUP] and [KEY-PAGEDOWN]. Linear or Sharp falloff works best with the sphere shape.

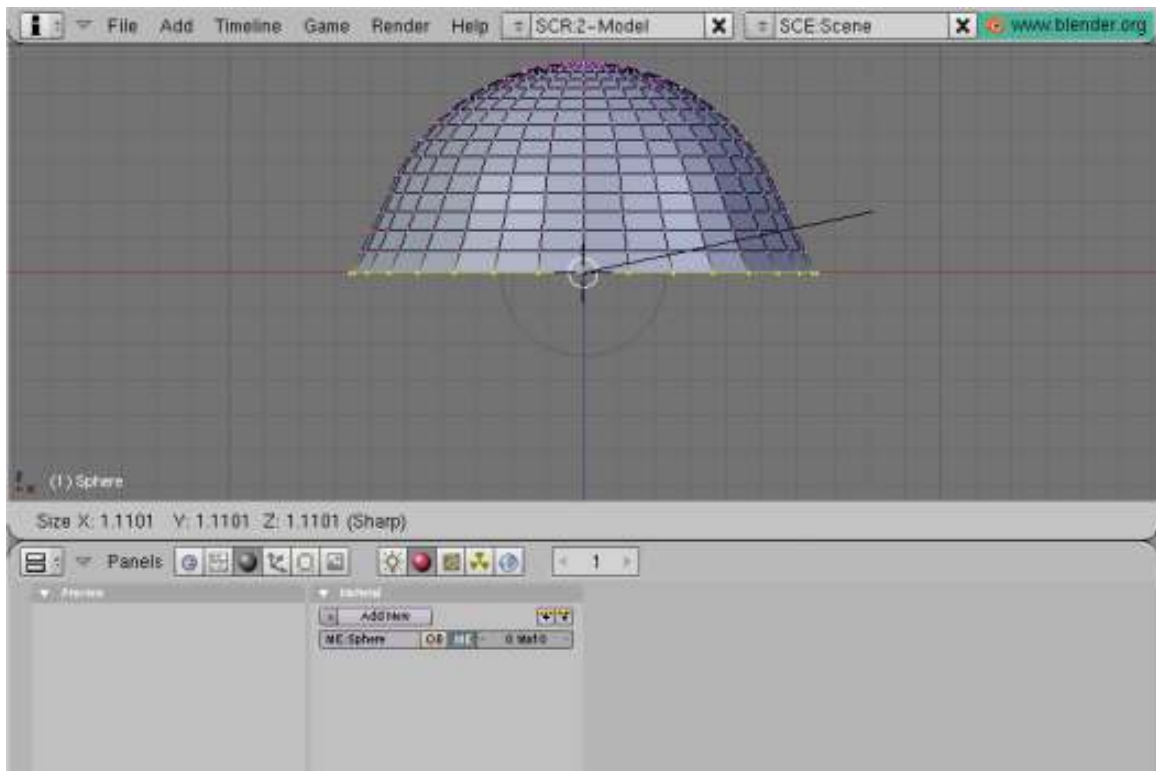


Figure 895 skybox tutorial 4

Now you're ready to add your sky texture to this mesh. In the Materials menu, create a new material and a new texture. Be sure to set your material not to receive shadows by clicking the "Shadeless" button. Then, in the Texture menu, set the texture type to Image, and click the Load Image button to insert our sky texture. Back in the Materials->Texture->Map Input menu, you may need to scale your image to get rid of the distorted textures at the edges of the fisheye by setting the Size to, say, 0.950 for X, Y and Z.



Figure 896 skybox tutorial 5



Figure 897 skybox tutorial 6

At this point, if you wish, you can reposition the camera and render the scene to see how your sky mesh looks.

115.3. Create a dome for the ground

I found it easiest to move the sky dome to a new layer with the [KEY-M] move to layer command. Then you can select the cube, Object->snap cursor to selection if you need to, select the top view [KEYPAD-7] and insert another UV sphere just as before -- except this time, remove the top hemisphere of vertices. I left an extra row of vertices at the "equator", scaled up, to function as a "billboard" to display the horizon of our ground texture with the alpha channel. This sphere should be slightly smaller than the sky hemisphere.

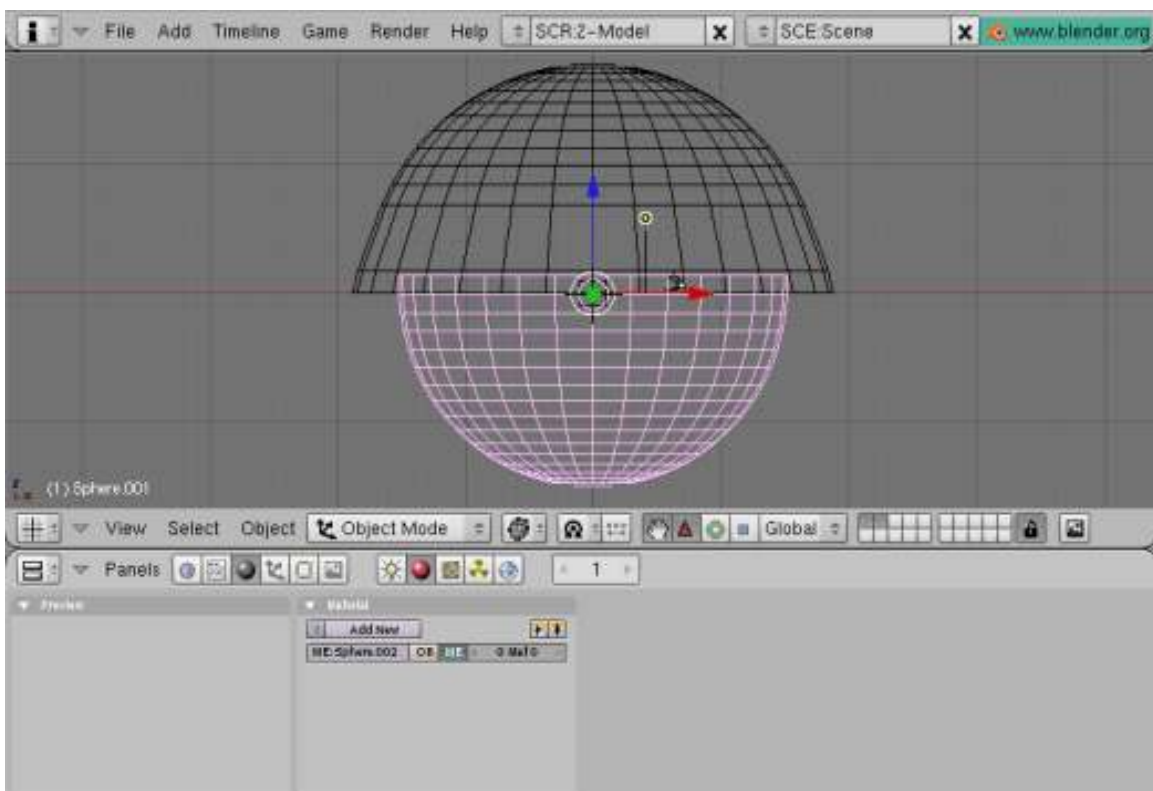


Figure 898 skybox tutorial 7

This time, I will apply the ground texture with a tube projection, so it is projected onto the mesh horizontally [Materials panel | Map Input tab | Tube button]. Because I have an alpha channel on this texture, I click "Use Alpha" in the Texture menu and Map To -> both Col and Alpha buttons [Materials Panel | Map To Tab | -> both Col and Alpha buttons]. You will also need to set ZTransp in the Mirror Transp menu [Materials Panel | Links and Pipelines Tab | ZTransp button] so that your alpha channel shows up in the envmap (which will become your skybox), and Alpha to 0 [Materials panel | Material tab | A slider] to allow the masked areas to be transparent. (Alpha channels appear to require Z buffering to appear on procedural textures.) Also, you may need to adjust the offset of the ground texture (Y-axis), so that the horizon appears properly on the "billboard" area of your ground hemisphere.



Figure 899 skybox tutorial 8



Figure 900 skybox tutorial 9

Again, you can reposition the camera and render the scene to make sure everything is properly aligned. Be sure to activate the layer where you moved the sky mesh. Your results will look similar to the following image. Set OSA on in the render screen for best results. Also, use higher resolution images with cleaner alpha channels -- the image below is rather blurry and you can see a halo around the horizon.

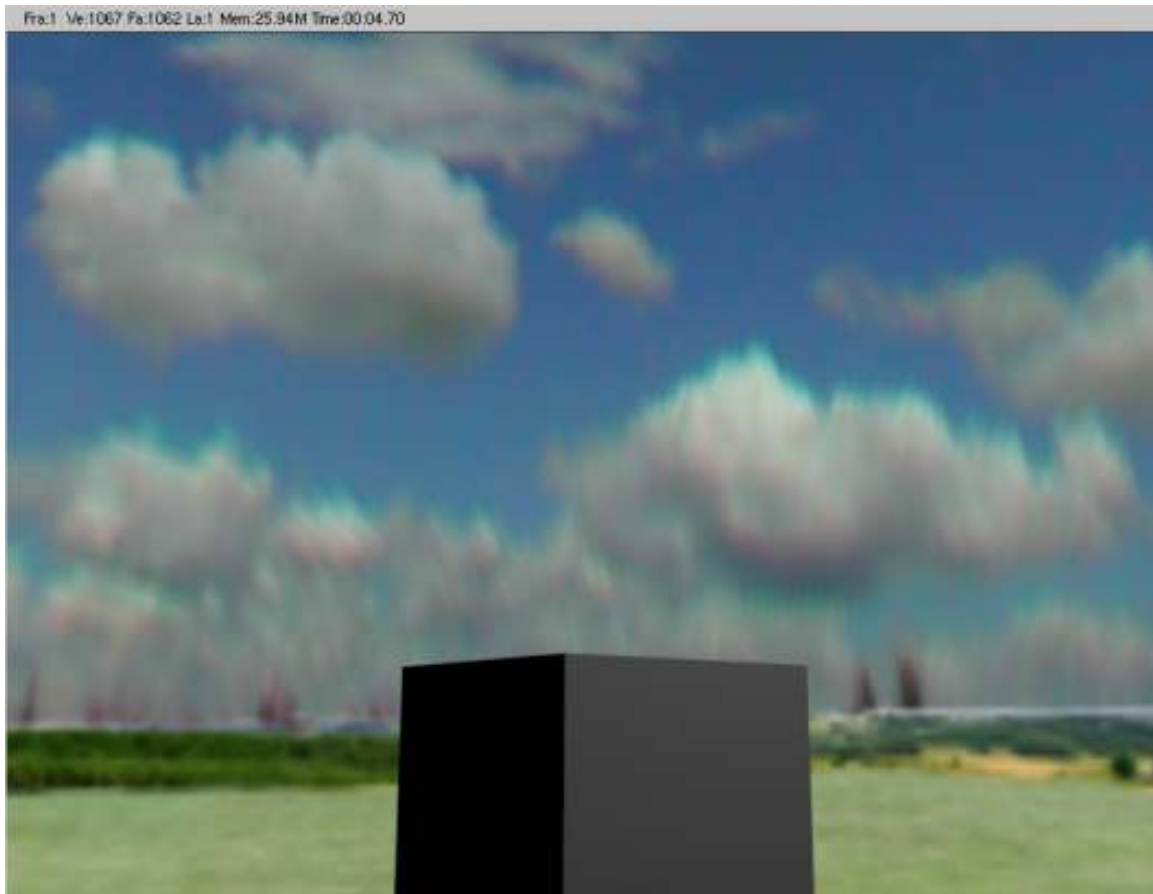


Figure 901 skybox tutorial 10

115.4. Render the environment map

The last step is to use the procedural Envmap texture to project the dome textures onto the cube, which will become our skybox. Select the cube and create a new material. Set the material to "Shadeless" [Materials Panel | Materials Tab | Shadeless button]. Add a new texture and make its type Envmap. Set the CubeRes [Envmap tab | CubeRes setting] to whatever you want the resolution of your skybox to be (512 is a good resolution for a game; 1024 or 2048 are fairly high-res; I stuck with low-res for this tutorial). If your sky & ground hemispheres are very physically large, you may also need to increase the ClipEnd value to include all of the faces. You may want to set the Envmap calculation to Anim so you don't have to keep freeing envmap data if you're experimenting. (Anim automatically clears Envmap data with every render, otherwise you must click 'Free Data' to reset the Envmap.)



Figure 902 skybox tutorial 11

Once you've created the Envmap texture, you should be ready to render the Envmap for your skybox. If you want to set your file format such as JPG or PNG, you should do that first. Then, simply go to the render screen and click "Render." Again, make sure all layers are visible. The rendering window appears. First, Blender renders the environment map of the cube. Afterward, the camera view is rendered, at which point you can hit [KEY-ESCAPE] to stop rendering -- we are only interested in the environment map which is already complete.

Select the cube again, then get to its texture menu. You will see the newly-rendered Envmap on the sample texture. Click "Save EnvMap" in the texture menu to save the rendered Envmap.



Figure 903 skybox tutorial 12

Blender environment maps are saved as a 3x2 matrix of squares, as seen here:



Figure 904 skybox tutorial 13

The cube faces are in the following order.

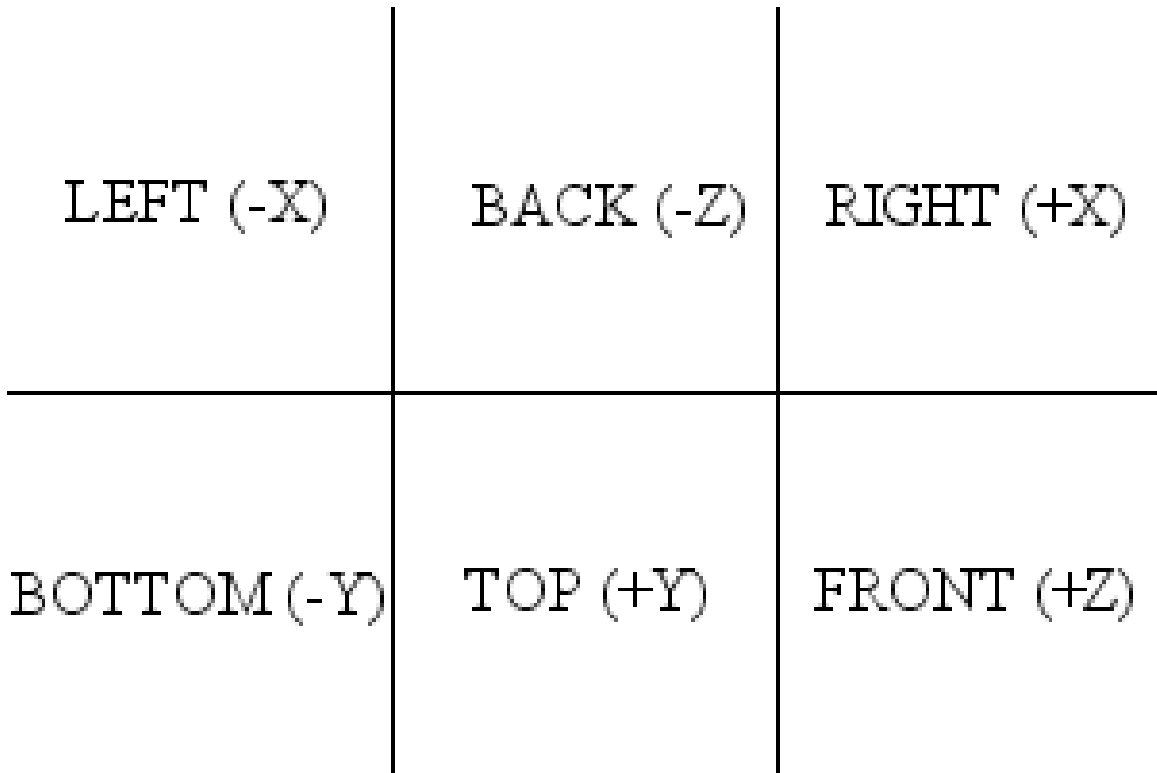


Figure 905 skybox tutorial 13

You can now load this image as an envmap texture in a new cube, which you can incorporate into your game as a skybox. This file can also be edited in the Gimp to remove any unwelcome artifacts such as trees, buildings, jet trails, etc. Also, because I used a tube projection on the lower hemisphere, in the bottom face of the envmap you see a strange star shape at the "pole." You'll most likely have a floor in your game, so you probably won't see that face anyway, but sticklers can avoid it with clever use of the Filters->Distorts->Polar Coords filter in the Gimp or Filter->Distort->Polar Coordinates (Polar to Rect.) in Photoshop. Patching also works well.

To make the skybox appear as a static background in your game, vertex-parent it to the current active camera object.

115.5. Video Tutorial

Ira Krakow's Blender 2.49 Skybox Tutorial: <http://www.youtube.com/watch?v=azkk3JrM5Es>

116. Basic mouse pointer

Making a simple mouse pointer in the game engine. This takes up a lot of resources but it is very simple.

A little Python is involved but it is very easy to use and is only 2 lines of code.

1. Open up blender and split the screen in two.
2. Make the right screen a text editor and add a new text file with **ALT+N**. Type in the following code.

```
import Rasterizer as r
r.showMouse(1)
```

3. Set *TX: showpointer* in the middle of the Text panel menu bar.
4. Select an object that will always be available - preferably a camera.
5. Go to the logic tab, add a "*Property*" sensor, a "*Python*" controller, an "*AND*" controller and a "*Property*" actuator.
6. Activate the *True Level Triggering* (the ““ button), set *Prop: switch* and *Value: 0*. Connect the property sensor to the python and AND controllers by dragging lines between the bullets.
7. For the python controller set *Script: showpointer*.
Note: If the value keeps being reverted to blank after setting it, the name you entered is not a legal script name; chances are you did not set the name of your script correctly. Look for the selection menu beginning with "TX:" and make sure it says *TX:showpointer*.
8. Now connect the AND controller to the property actuator. Set *Prop: switch* and *Value: 1*.
9. Select *Actor* and click on *Add Property* make it a *Int* type and set *Name:switch*.

Now press **P** to start the game and now you'll see your mouse pointer.

117. Text in BGE

There are a lot of tutorials that show how to make text for the Blender Game Engine, for example to use in menus. Most involve editors, graphics programs, targa files, UV mapping, scripting, higher magic and so on. Here is a different approach that takes but a minute.

1. Start Blender.
2. Remove the default cube. (**X**)
3. Add text. (**SPACE** → *Add* → *Text*)
4. You may want to switch to edit mode (**TAB**) and use the variety of features that Blender provides for editing, formatting and laying out text. They are described in the Blender Manual Section¹. For a start just use **BACKSPACE** to delete the letters "Text" and type your own like "This is simple".
5. When finished editing go back to object mode. (**TAB**)
6. Convert your text to a mesh. (**ALT+C** → *Mesh*)
7. Press **P**

This is as simple as it gets. Of course you can do all kinds of laying out with your text before the conversion in step 6 (check the Blender Manual) as well as modeling, texturing and dynamically manipulating thereafter (as you have a normal mesh). Simply remember that you need to make sure the text is final before you convert it.

This tutorial has been tested with Blender 2.8 RC1.

"However, it should be pointed out that while this is so incredibly simple, it is not poly-count efficient. Remember, there is more to making a game than making it look good--making it run well. If you need a cut-and-dry menu, this is the best way. If you need to worry about performance, then it might be worth it to check out some UV Mapping and external program usage. --Bullercruz1² (talk³) 23:07, 7 February 2009 (UTC)"

It is also important to note that these fonts can't be dynamic, contrary to bitmap-uvmapped ones...

- How to create Dynamic Text in Blender Game Engine (Youtube)⁴

1 <http://wiki.blender.org/index.php/Manual/Text>

2 <http://en.wikibooks.org/wiki/User%3ABullercruz1>

3 <http://en.wikibooks.org/wiki/User%20talk%3ABullercruz1>

4 http://www.youtube.com/watch?v=kcGLy1__QsA

118. Python Platformer: Creation

Note: Python code is placed in a Text Editor window. It might be helpful to split your 3D View window into a separate part, so that you can use a buttons, 3D view, and text editor window simultaneously.

The initial stages of the Python Platformer tutorial series will mostly have to do with replicating what was done in the logic board Platformer tutorials. If you have not read and understood those tutorials, you may not understand exactly what the function of most of this code is. This tutorial details:

1. An explanation of the "code sections" to be modified, rewritten, or redefined frequently throughout the series
2. Creating an object linked to a pre-made mesh
3. Linking an object to a scene

and will detail by completion:

1. How to move an object in response to keyboard triggers

118.1. Code Sections

We will refer to the sections of the code in this manner throughout the series:

118.1.1. Import Section

The series of import commands at the beginning of the piece of code, like the contents of the head tag in HTML documents. Python has a basic set of commands kept naturally in the language, and the rest are imported so that a large amount of commands aren't loaded when not needed, causing unnecessary memory expenditure. This lends the advantage of being able to extend the language by writing custom sets of commands to be imported into Python, which is how Blender interfaces with Python. For this tutorial, our import section is:

```
import Blender
import bpy
```

When you import something as something else, it basically creates a variable equal to the imported module's name, so that you don't have to type it out. It is inessential, but it makes the coding go faster. You might want to run a find and replace search on your document afterwards, replacing all instances of the name you imported it as with the real module's name and deleting the part that imports the module by another name, as this can make the code run slightly faster.

118.1.2. Essential Footer

This is the collection of commands that you must remember to include at the end of the document, regardless of any changes made to the code, for it to work. It must be noted whether you should change the entire footer (as if a feature was removed from the main code-in-progress) or adding a command to your existing footer (if the text deals with the addition of a single feature). Every article in the main tutorial line should contain the full footer as it should look at that point in the series. For this tutorial, our essential footer is:

```
Blender.Redraw()
```

118.2. Adding the Player Object

To start, create a new document and delete the basic cube. Create a Monkey/Suzanne. In the Editing panel with the monkey selected, change the mesh name (ME:) under Link and Materials to "Hero". Click the F next to the input box to preserve the data block even when nothing links to it. Now delete the monkey object.

The following code will create an object called "Player" in the library and link it to the Hero mesh:

```
player = Blender.Object.New("Mesh", "Player")
player.link(bpy.data.meshes["Hero"])
```

In the first line, `Blender.Object.New` obviously references a new object. The "Mesh" variable should not be changed for the purposes of this code. I don't know exactly it's function, so I don't want to give out misinformation, but I speculate that it has to do with the object type. If the object were to be a lamp or camera type, for example, you would not be able to apply a mesh to it. The second variable, "Player", is the name of the actual object you're creating. Change it to your liking.

In the second line, we link `player` (which was equated to our new object) to the pre-existing mesh "Hero", which is the Suzanne mesh we dealt with before. Using this method, you can model your character's mesh beforehand but have the actual character created dynamically. Using a complication of this code, you can make `player.link()` link to a variable and not the bit "`bpy.data.meshes[]`". That variable can reference an existing mesh or it can create a new one. As for the meaning of the link's value, `bpy.data.meshes` is an array containing all the meshes in the movie. Likewise, `bpy.data.objects` contains all the objects in the movie. By going to the Scripts window and going to **Scripts->System->Interactive Console** you can gain such information as the contents of these arrays. By entering "`list(bpy.data.objects)`" into the console, you will be rewarded with a list in the format of `[[Object "Camera"], [Object "Cube"], [Object "Lamp"]]` which is the list of objects in the standard new document set-up. So, to reference the item `[Object "Cube"]` you would use the line "`bpy.data.objects['Cube']`" and so on. For these commands to work, you must make sure to import `bpy` in your import section.

118.2.1. Appending the Object to the Scene

Like you must append the mesh to the object, you must also append the object to the scene, or it will just be a floating data block and not actually appear anywhere.

```
scene = Blender.Scene.GetCurrent ()  
scene.link(player)
```

If you were to test this code by pressing **ALT-P** while the mouse is hovering over the Text Editor window, it would create an object named Player with the monkey "Hero" mesh at the origin point of your scene. Make sure you included this tutorial's import section before all of the other code and the essential footer after all of other code.

Part IV.

Unit 4 - Taking Off With Advanced Tutorials

119. Python Scripting

Information

Applicable Blender version: 2.49

One of Blender's powerful features is its Python API. This allows you to interface with Blender through the Python programming language. The Python interface allows you to control almost all aspects of Blender, for example you can write import or export scripts for meshes and materials of various formats or create procedurally generated textures. You can also create complete animations procedurally and write scripts to modify existing scenes in any way you can think of. On top of all, you can easily create a user interface for your script, transforming it into a generally usable tool.

The relevant section of the Blender User's Manual is [here](#)¹.

Download Python Scripts from the Blender site [here](#)².

This forum³ on blenderartists.org discusses Python scripting.

here is a remix and update to the great noob tutorial : /Tutorial (Blender 2.4.9)/⁴

119.1. Chapter contents

1. Introduction⁵
2. Procedural object creation⁶
3. External scripts⁷
4. Creating a GUI for your script⁸
5. Export scripts⁹
6. Import scripts¹⁰
7. Scripts for modifying meshes¹¹

1 http://wiki.blender.org/index.php/Manual/Python_Scripting

2 <http://wiki.blender.org/index.php/Extensions:Py/Scripts/Catalog>

3 <http://blenderartists.org/forum/forumdisplay.php?f=11>

4 <http://en.wikibooks.org/wiki/%2FTutorial%20%28Blender%202.4.9%29%2F>

5 Chapter 120 on page 989

6 Chapter 127 on page 1033

7 Chapter 128 on page 1037

8 Chapter 129 on page 1041

9 Chapter 130 on page 1045

10 Chapter 131 on page 1055

11 Chapter 132 on page 1057

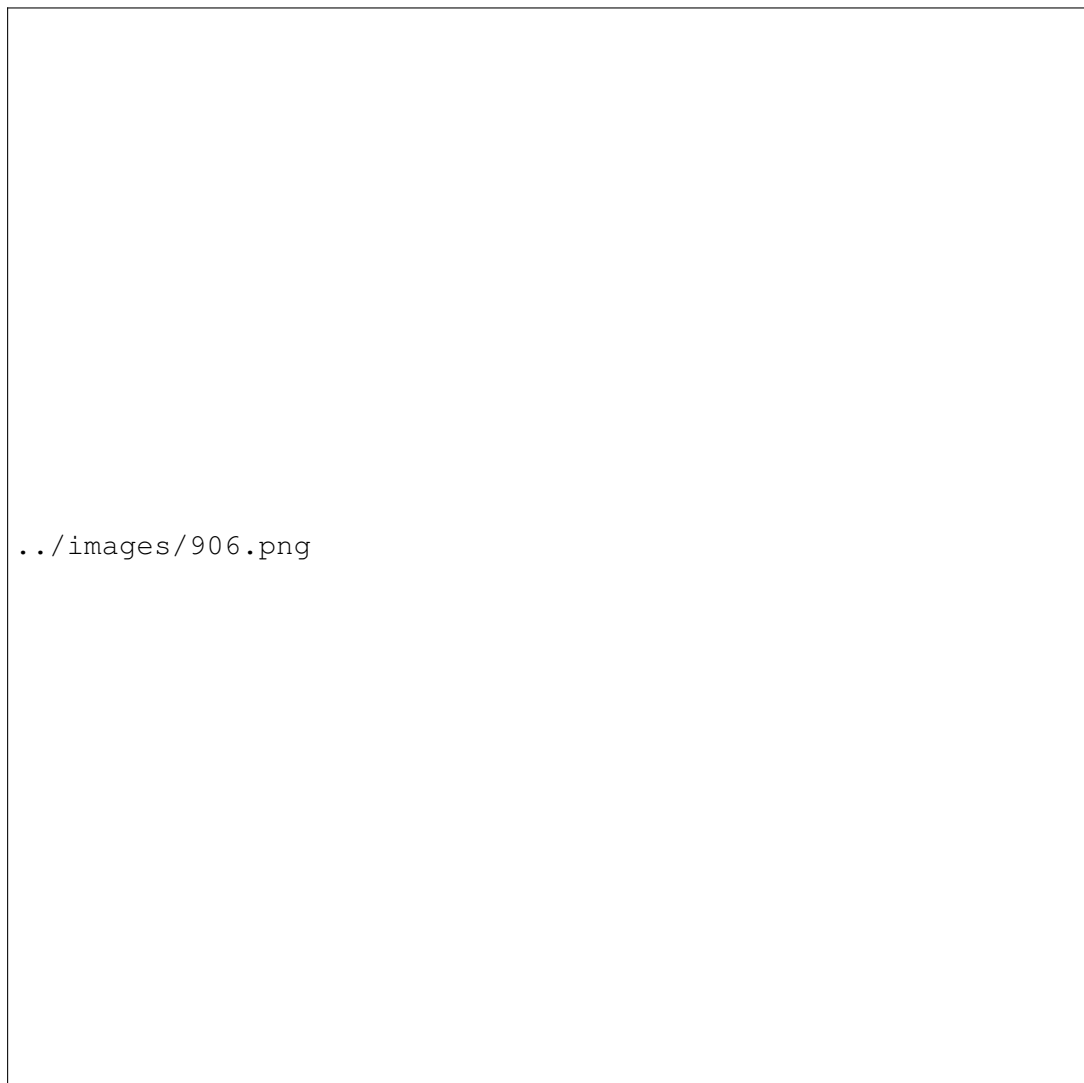


Figure 906

Note:

Scripting in the upcoming version of Blender (currently in development as the version 2.5x series) is completely different, and is described here^a.

^a Chapter 121 on page 993

120. Introduction

i Information

Applicable Blender version: 2.49

120.1. Introduction

Python is an incredible language, much akin to C, which also offers linking of C libraries, probably why it was chosen to be integrated into Blender. It was designed to incorporate the extremely quick and easy learning of ABC with the more powerful functionality of C, but at the same time keeping syntaxing and naming conventions simple.

Arguably its main function is in importing and exporting 3D models, an essential task for a 3D modeling application. With the adaptability of Python, it allows you to create custom import/export scripts easily and quickly for Blender, to service filetypes which Blender doesn't natively support.

Next comes its ability to allow automation, such as the Discombobulator, which is a fantastic (but not often used) tool, which turns simple meshes into futuristic panels.

It is also almost universally used in the BGE (Blender Game Engine), to bring life to games.

And yes, Python was named after *Monty Python*.

The version of Python used with Blender 2.49 is 2.6. If you are unfamiliar with Python, start with this tutorial¹ at python.org.

120.2. First Steps In Blender Scripting

Open a new, default Blender document. Split the 3D View in two vertically. Change the type of one side to a Scripts Window. In the “Scripts” menu in the header of that window, find the “System” submenu, and select “Interactive Python Console”. The grey background should turn black, with some initial explanatory blue text; everything you type into this window will display in white, while output from the system will be in blue.

To start with, let's find out what objects are present in the document. Type

```
Blender.Object.Get()
```

The response should be

¹ <http://docs.python.org/tutorial/>


```
[[Object "Camera"], [Object "Cube"], [Object "Lamp"]]
```

which shows you the names of the three default objects you can see in the 3D View window.

Let's get a reference to the Cube object for more convenient access: type

```
Cube = Blender.Object.Get("Cube")
```

Now let's try querying the value of one of its attributes: type

```
Cube.LocX
```

You should see the response

```
0.0
```

Now let's try changing its location: type

```
Cube.LocX += 1
```

Nothing seems to happen. But now tell Blender to update its display:

```
Blender.Window.Redraw()
```

and you should see the cube jump to a new position in the 3D View. To make it move again, press **UPARROW** a couple of times to bring back the command that changed the LocX attribute, followed by **ENTER** to re-execute it; then do the same with the Redraw command. And that's it—you're scripting!

120.3. Multiple Running Scripts

After starting the console, you should see a popup menu in the Script Window header showing “Interactive Python C”, being the truncated name of the script you started running. You can have multiple scripts running at once, and switch between them by selecting from this menu; for example, try starting a second Python console by re-executing the “Interactive Python Console” script. To terminate the currently-running script, press **CTRL** + **Q** in this window.

120.4. The Blender Versus bpy Modules

The main functionality that Blender makes available to Python scripts is in the Blender module and its submodules. However, beginning with Blender 2.44, a new bpy module was introduced. This represents a completely revamped way of accessing Blender's functionality, which maps in a more natural fashion onto Python language constructs.

Indeed, in the upcoming 2.5x version of Blender (still under development), the Blender module is abandoned completely, and what equivalent functionality there is must be accessed through bpy.

So, does this sound like a good reason to start using module bpy in preference to module Blender? Unfortunately, in the 2.4x series, the bpy module is still marked as “experimental”, and its functionality is far from complete.

In short, there is no way to write scripts for Blender 2.49 so that they will also work seamlessly with Blender 2.5x.

120.5. See Also

Python API Reference²

² <http://www.blender.org/documentation/249PythonDoc/index.html>

121. Introduction (version 2.5)

Python is a powerful, high-level, dynamic language. The version of Python used in Blender 2.5x is version 3.1. If you are unfamiliar with Python, start with [this tutorial](#)¹ at python.org. If you are familiar with older (2.x) versions of Python, [this page](#)² summarizes what's new in 3.x.

If you are familiar with Python scripting in older versions of Blender, be aware that 2.5x is completely different; the old Blender module is gone. Much of the functionality is now available through the `bpy` module, but don't expect an exact 1:1 correspondence.

121.1. First Steps In Blender Scripting

Open a new, default Blender document. If you haven't customized your settings, there will be a Timeline window along the bottom; change this to a Python Console. Perhaps increase its height to let you see more lines at once.

To start with, let's find out what objects are present in the document. At the ">>>" prompt, type

```
bpy.data.objects
```

You should see the response

```
<bpy_collection[3], BlendData.objects>
```

which isn't actually all that informative. In fact what you have here is an *iterator*; to see its contents, just do the usual Python thing and convert it to a list. Try entering this:

```
list(bpy.data.objects) #or being lazy: bpy.data.objects[:]
```

This time the response should be

```
[bpy.data.objects["Camera"], bpy.data.objects["Cube"],  
bpy.data.objects["Lamp"]]
```

which shows you how to refer to the three default objects you can see in the 3D View window.

Let's get a reference to the Cube object for more convenient access: type

```
Cube = bpy.data.objects["Cube"]
```

Now let's try querying the value of one of its attributes: type

¹ <http://docs.python.org/py3k/tutorial/>
² <http://docs.python.org/py3k/whatsnew/3.0.html>

```
Cube.delta_location
```

You should see the response

```
Vector((0.0, 0.0, 0.0))
```

The `Vector` type comes from the `mathutils` module provided by Blender. But unlike `bpy`, this is not automatically imported into the Python Console. So let's bring it in for subsequent use: type

```
import mathutils
```

OK, now let's try changing the location of the default Cube: type

```
Cube.delta_location += mathutils.Vector((1, 1, 1))
```

(Note the doubled parentheses: `mathutils.Vector` takes a *single* argument, which is a tuple of X, Y and Z coordinate values.)

Were you watching the 3D View when you pressed `[ENTER]`? You should have seen the cube jump to a different position. To make it move again, press `[UPARROW]` to bring back the above command, so you can execute it again with `[ENTER]`. As soon as you do this, the cube will jump another step, like before. And that's it—you're scripting!

121.2. The bpy Module

The contents of the `bpy` module are divided into several submodules (actually classes, but never mind), among which are:

- `bpy.data` — This is where you find the contents of the current document.
- `bpy.types` — information about the types of the objects in `bpy.data`.
- `bpy.ops` — *operators* perform the actual functions of Blender; these can be attached to hotkeys, menu items and buttons. And of course they can be invoked from a script. When you write an addon script, it will typically define new operators. Every operator must have a unique name.
- `bpy.context` — contains settings like the current 3D mode, which objects are selected, and so on. Also accessible in the Console window via the global variable `C`.
- `bpy.props` — functions for defining *properties*. These allow your scripts to attach custom information to a scene, that for example the user can adjust through interface elements to control the behaviour of the scripts.

121.3. The mathutils Module

The module `mathutils` defines several important classes that are used heavily in the rest of the Blender API.

- `Vector`: the representation of 2D or 3D coordinates.
- `Matrix`: the most general way of representing any kind of linear transformation.

- Euler: a straightforward way of representing rotations as a set of *Euler angles*, which are simply rotation angles about the X, Y and Z axes. Prone to well-known pitfalls such as gimbal lock³.
- Quaternion: on the face of it, a more mathematically abstruse way of representing rotations. But in fact this has many nice properties, like absence of gimbal lock, and smoother interpolation between two arbitrary rotations. The latter is particularly important in character animation.
- Color: a representation of RGB colours and conversion to/from HSV space (no alpha channel).



121.4. See Also

- Blender 2.5x API Reference⁴

³ http://en.wikipedia.org/wiki/Gimbal_lock

⁴ http://www.blender.org/documentation/blender_python_api_2_56_0/

122. Addon Anatomy (version 2.5)

An *addon* is a Python script that can extend the functionality of Blender in various ways. It can reside in a file within your Blender user preferences directory, or it can be stored in a text block within the Blender document. In the former case, the addon needs to be *enabled* in each Blender document where you want to use it, by ticking the checkbox in its entry in the Add-Ons list in the User Preferences window for that document. In the latter case, the script can be run by pressing  +  in the Text Editor window; or you can tick the “Register” checkbox for that script in the Text Editor, to have the addon be automatically enabled as soon as the document is loaded into Blender.

The main thing an addon script typically does is define one or more new *operators*. All the user-interface functions in Blender are performed by operators; these are invoked by being associated with menu items, buttons or hotkeys. Each operator is a subclass of the `bpy.types.Operator` class. The name you give your class in Python is immaterial outside the script where it is defined; it will be referred to from the rest of Blender via its *operator name*, which must be unique within the document. In the same way, your script can invoke other operators defined elsewhere via Blender API functions that take their operator name.

122.1. Text Blocks

A Blender document can contain *text blocks*, which are not the same as *text objects* in a 3D scene (though the former can be converted to the latter). Besides generating text objects, a text block can serve any purpose you like; for example, use it to pass workflow instructions to a colleague along with the document; display a copyright or help message in the initial layout that a user sees on opening the document; or hold a Python script that can be run by the user to perform some useful action related to the document.

122.2. Your First Operator

Open a new, empty Blender document. Bring up the Text Editor in any convenient window; you will see an empty, grey rectangle. Before you can type in text, you need to create a text block; do so by clicking the large button labelled “New” in the window header. As soon as you do this, you should see a popup menu appear listing all the text blocks in your document, with the current (only) entry named “Text”. You should also see a red insertion cursor appear at the top left, indicating that you can start typing.

Unlike the Console Window, nothing is automatically imported for you. So as in any other Python script, you need to mention every module you want to access.

You define your operator as a subclass of a subclass of the `bpy.types.Operator`. It must have a `bl_idname` class attribute which gives the operator name, while `bl_label` gives the user-visible name

that appears in the spacebar menu. The former must have valid Python syntax for a name, including a single dot; the part to the left of the dot must be the name of one of the valid categories of operators, which you can find by typing

```
dir(bpy.ops)
```

into the Console Window.

Let's define an operator that will add a new tetrahedron object to the document. The class definition should start something like this:

```
class MakeTetrahedron(bpy.types.Operator) :
    bl_idname = "mesh.make_tetrahedron"
    bl_label = "Add Tetrahedron"
```

It has to define an invoke method, which starts off like this:

```
def invoke(self, context, event) :
```

The invoke method performs the actual function of the operator; after it has finished, it must return a value which is a set of strings, telling Blender things like whether the operator is still executing in a modal state, or whether the operation has finished, and if so whether it was successful or not. To indicate successful completion: end the invoke method with this:

```
return {"FINISHED"}
```

Anyway, let us compute the coordinates of the vertices for the tetrahedron: with edges with a length of 1 Blender unit, suitable values are $(0, -\frac{1}{\sqrt{3}}, 0)$, $(\frac{1}{2}, \frac{1}{2\sqrt{3}}, 0)$, $(-\frac{1}{2}, \frac{1}{2\sqrt{3}}, 0)$ and $(0, 0, \sqrt{\frac{2}{3}})$. Or in Python:

```
Vertices = \
[
    mathutils.Vector((0, -1 / math.sqrt(3), 0)),
    mathutils.Vector((0.5, 1 / (2 * math.sqrt(3)), 0)),
    mathutils.Vector((-0.5, 1 / (2 * math.sqrt(3)), 0)),
    mathutils.Vector((0, 0, math.sqrt(2 / 3))),
]
```

Then we create the mesh datablock, giving it the name "Tetrahedron":

```
NewMesh = bpy.data.meshes.new("Tetrahedron")
```

Fill it in with the above vertex definitions and associated faces:

```
NewMesh.from_pydata \
(
    Vertices,
    [],
    [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
)
```

The Mesh.from_pydata method is not currently well documented in the Blender API, but its first argument is an array of Vector defining the vertices, the second argument is an array of edge definitions, and the third argument is an array of face definitions. Each edge or face is defined as a list of vertex indexes (2 elements in each list for an edge, 3 or 4 for a face), 0-based in usual Python style, being indexes into the array of vertex definitions that you passed. Note that you pass **either the**

edge definitions or the face definitions, but not both: the other one should be passed as the empty list. If this function is not used correctly, it will cause Blender to crash.

We also need to add the following, to tell Blender the mesh has changed and needs updating (as if it couldn't figure it out itself):

```
NewMesh.update()
```

(Omitting this produces a minor quirk: right-clicking the newly-created tetrahedron does not highlight its outline in orange as with other objects, though the problem goes away if you enter edit mode on the object and exit again.)

Now to create the object datablock (I also give it the name "Tetrahedron"), and link it to the mesh:

```
NewObj = bpy.data.objects.new("Tetrahedron", NewMesh)
```

But the object will not appear to the user until it is linked into the scene:

```
context.scene.objects.link(NewObj)
```

To recap, here is the complete script:

```
import math
import bpy
import mathutils

class MakeTetrahedron(bpy.types.Operator) :
    bl_idname = "mesh.make_tetrahedron"
    bl_label = "Add Tetrahedron"

    def invoke(self, context, event) :
        Vertices = \
            [
                mathutils.Vector((0, -1 / math.sqrt(3), 0)),
                mathutils.Vector((0.5, 1 / (2 * math.sqrt(3)), 0)),
                mathutils.Vector((-0.5, 1 / (2 * math.sqrt(3)), 0)),
                mathutils.Vector((0, 0, math.sqrt(2 / 3))),
            ]
        NewMesh = bpy.data.meshes.new("Tetrahedron")
        NewMesh.from_pydata \
            (
                Vertices,
                [],
                [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
            )
        NewMesh.update()
        NewObj = bpy.data.objects.new("Tetrahedron", NewMesh)
        context.scene.objects.link(NewObj)
        return {"FINISHED"}

#end invoke
bpy.utils.register_class(MakeTetrahedron)
#end MakeTetrahedron
```

Now execute your script by typing **ALT** + **P**. If all goes well *nothing should appear to happen*; Blender will see the Python class definition, note that it is an operator, and automatically register it, ready for use.

If you hit any syntax errors, Blender should display these in a popup window; go back and correct them, and reexecute the script with **ALT** + **P**.

122.3. Invoking Your Operator

We haven't (yet) defined any user interface for this operator; so how do we invoke it? Simple.

Go to a 3D View window. Delete the default cube to avoid it obscuring things, and press **[SPACE]**. This brings up a searchable menu of every operator defined for the current document. Into the search box, type part or all of the string you defined for the `bl_label` attribute above (typing "tetra" should probably be enough). This will restrict the menu to only showing items that contain that string, which should include the name of your operator; click with **[LMB]** on this or highlight it and press **[ENTER]**. If all goes well, you should see your tetrahedron object appear!

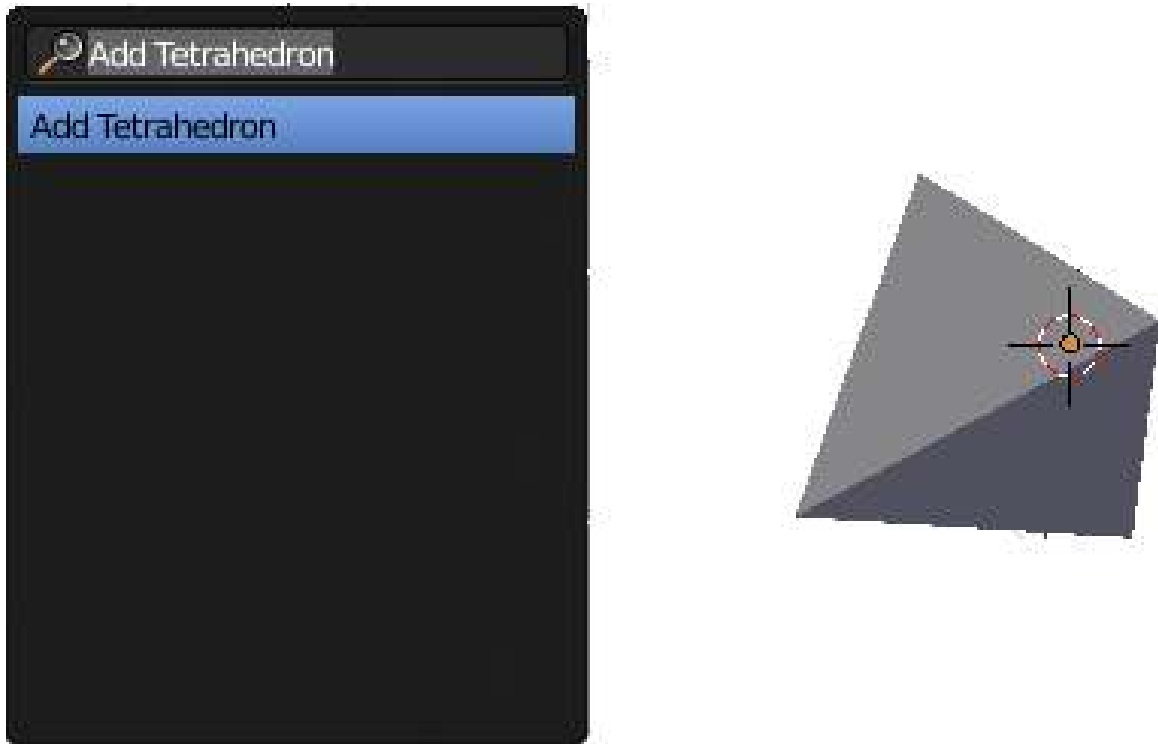


Figure 907

Script does not invoke operator for Blender v.2.57 r36339M (current release as at May 28, 2011). Add the following at foot of py script: `bpy.utils.register_class(MakeTetrahedron)`

122.4. If You Hit An Error

If there is any error running the script, Blender should display this in a popup. For example, the following simple one-line script

```
raise RuntimeError("Uh-oh")
```

displays this message:

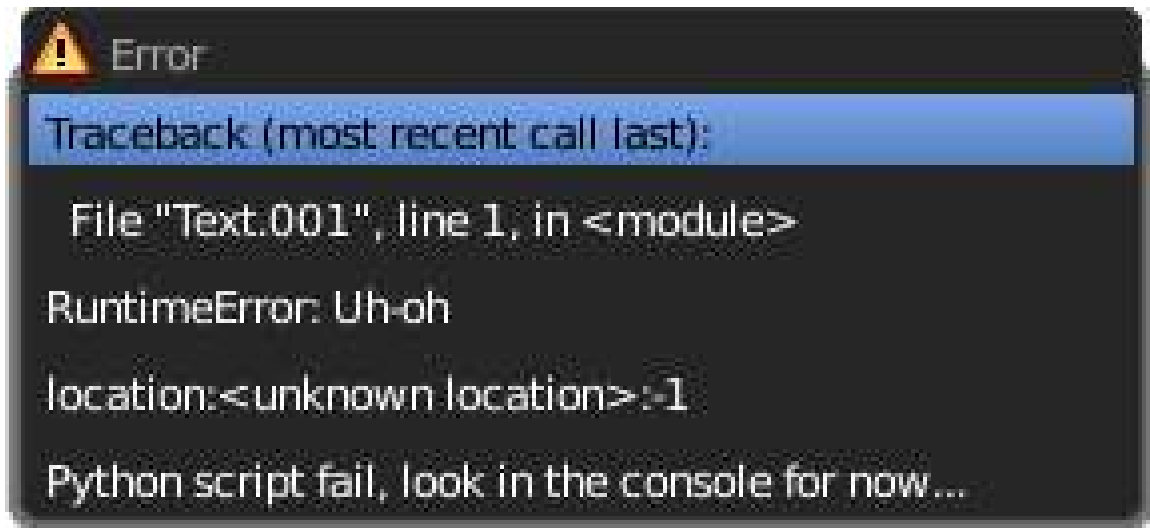



Figure 908

The Python traceback message is also written to Standard Error; on Linux/Unix systems, the message will appear in the terminal session if you invoked Blender from the command line; otherwise it will be appended to your `~/.xsessionerrors` file if you launched Blender from a GUI. On Windows the message appears in the console window.

123. Addon User Interface (version 2.5)

So you've defined a new operator in your addon. Now it would be nice to give that addon a proper interface, so the user doesn't have to hunt through the spacebar menu to find that operator.

The nicest way to do this is to define a *panel* that shows up in a window somewhere, with controls in it that the user can click on to operate your addon. You can put the panel in all kinds of places, but here we will insert it in the Tool Shelf (which can be hidden or shown on the left side of the 3D View by pressing ).

123.1. Your First Panel

A panel is defined by subclassing the `bpy.types.Panel` class. You set the value of various attributes (`bl_space_type`, `bl_region_type` and `bl_context`) to determine the context in which the panel will appear, and also give a title to the panel (`bl_label`):

```
class TetrahedronMakerPanel(bpy.types.Panel) :
    bl_space_type = "VIEW_3D"
    bl_region_type = "TOOLS"
    bl_context = "objectmode"
    bl_label = "Add Tetrahedron"
```

Those first three attributes cause the panel to appear in the Tool Shelf, but only while the 3D View is in Object mode.

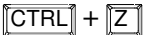

Your class also needs to define a `draw` method, which defines the items that go into the panel. This example creates a new column of UI elements for the panel, and inserts a single UI element which is a button that will invoke the operator with the name you previously defined when clicked:

```
def draw(self, context) :
    TheCol = self.layout.column(align = True)
    TheCol.operator("mesh.make_tetrahedron", text = "Add
Tetrahedron")
#end draw
```

123.2. Adding Undo Support

While we're at it, go back and add the following line to the operator definition:

```
bl_options = {"UNDO"}
```

This allows the user to undo the addition of the tetrahedron in the usual way with , and redo it with .

123.3. Put It All Together

Here is the complete script for the addon as it stands now:

```
import math
import bpy
import mathutils

class TetrahedronMakerPanel(bpy.types.Panel) :
    bl_space_type = "VIEW_3D"
    bl_region_type = "TOOLS"
    bl_context = "objectmode"
    bl_label = "Add Tetrahedron"



    def draw(self, context) :
        TheCol = self.layout.column(align = True)
        TheCol.operator("mesh.make_tetrahedron", text = "Add
Tetrahedron")
        #end draw

#end TetrahedronMakerPanel

class MakeTetrahedron(bpy.types.Operator) :
    bl_idname = "mesh.make_tetrahedron"
    bl_label = "Add Tetrahedron"
    bl_options = {"UNDO"}

    def invoke(self, context, event) :
        Vertices = \
            [
                mathutils.Vector((0, -1 / math.sqrt(3),0)),
                mathutils.Vector((0.5, 1 / (2 * math.sqrt(3)), 0)),
                mathutils.Vector((-0.5, 1 / (2 * math.sqrt(3)), 0)),
                mathutils.Vector((0, 0, math.sqrt(2 / 3))),
            ]
        NewMesh = bpy.data.meshes.new("Tetrahedron")
        NewMesh.from_pydata \
            (
                Vertices,
                [],
                [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
            )
        NewMesh.update()
        NewObj = bpy.data.objects.new("Tetrahedron", NewMesh)
        context.scene.objects.link(NewObj)
        return {"FINISHED"}
    #end invoke

#end MakeTetrahedron
```

As before, hit  +  to execute it. Nothing should appear to happen; Blender notices your subclass definitions, and automatically registers them for use in the appropriate places.


But now, go back to the 3D View. Make sure you are in Object mode. Bring up the Tool Shelf if it isn't visible, by pressing . At the bottom, you should see your new panel appear:



Figure 909

Note the triangle next to the title that Blender automatically gives you, to collapse or expand the panel without any extra work on your part. Get rid of any tetrahedron object that might have been created by following the tutorial on the previous page; now click your new “Add Tetrahedron” button, and watch the object be created again!

For Blender v.2.57 r36339M add to py script at foot

```
bpy.utils.register_class(MakeTetrahedron)
```

```
bpy.utils.register_class(TetrahedronMakerPanel)
```

123.4. Space Types, Region Types, Contexts, Oh My!

Those values for `bl_space_type`, `bl_region_type` and `bl_context` aren't currently explained anywhere in the Blender API documentation. So here's a list of what I've been able to find from looking at source code:

Attribute	Permissible Values
<code>bl_space_type</code>	"EMPTY", "VIEW_3D", "GRAPH_EDITOR", "OUTLINER", "PROPERTIES", "FILE_BROWSER", "IMAGE_EDITOR", "INFO", "SEQUENCE_EDITOR", "TEXT_EDITOR", "AUDIO_WINDOW", "DOPESHEET_EDITOR", "NLA_EDITOR", "SCRIPTS_WINDOW", "TIMELINE", "NODE_EDITOR", "LOGIC_EDITOR", "CONSOLE", "USER_PREFERENCES"
<code>bl_region_type</code>	"WINDOW", "HEADER", "CHANNELS", "TEMPORARY", "UI", "TOOLS", "TOOL_PROPS", "PREVIEW"

Attribute	Permissible Values
bl_context	"mesh_edit", "curve_edit", "surface_edit", "text_edit", "armature_edit", "mball_edit", "lattice_edit", "posemode", "sculpt_mode", "weightpaint", "vertexpaint", "texturepaint", "particlemode", "objectmode"

Of course, not all combinations will make sense.

See also; http://www.blender.org/documentation/blender_python_api_2_57_release/bpy.types.Panel.html¹

¹ http://www.blender.org/documentation/blender_python_api_2_57_release/bpy.types.Panel.html

124. Custom Property (version 2.5)

Addons will commonly do more than unconditionally perform a single action; the user may be able to control their actions in various ways. We can add controls for adjusting settings to our addon panel, and associate these with custom *properties* that we define for the current scene; Blender will then take care of updating the property values as the user operates those controls, and our operator's invoke routine can fetch those property values when it runs.

124.1. Defining A Property

Properties need to be defined in our *register* method. Our addon doesn't currently have one, so let's add it:

```
def register() :
    bpy.types.Scene.make_tetrahedron_inverted =
    bpy.props.BoolProperty \
    (
        name = "Upside Down",
        description = "Generate the tetrahedron upside down",
        default = False
    )
#end register
```

Here we are attaching the property as a new attribute of Blender's Scene class; Python lets us assign new attributes to just about any object, and Blender takes full advantage of that. Note that the property must be created using one of the property-definition routines provided in `bpy.props`: choose the one that matches the type of property you want to create. Here we are defining a simple true/false toggle, which the user will control via a checkbox. Whatever name you use for your custom class attribute, an instance of that class will have an attribute with the same name, holding the actual value for that property.

The name will be used as the name of a control for examining or changing this property, while the description will appear as a tooltip when the user hovers the mouse over the control. The default is used as the initial value of the property.

Note also I tried to use a name, `make_tetrahedron_inverted`, which is less likely to clash with names defined by other addons or parts of Blender.

Let's also add an *unregister* method. This won't actually be used for now, but it will become relevant later when we extract the addon for separate installation:

```
def unregister() :
    del bpy.types.Scene.make_tetrahedron_inverted
#end unregister
```

To make the checkbox appear, add the following line to our panel's draw routine:

```
TheCol.prop(context.scene, "make_tetrahedron_inverted")
```

The first argument to the prop method must be an instance of the class to which we attached our property definition above; in this case, it is the current scene.

Finally, all Blender's smarts about automatically registering our Panel and Operator subclasses doesn't extend to finding and executing the register routine (not in the Text Editor, anyway). So we need to finish off our script with the following bit of boilerplate which will handle the custom registration routine in the situations where Blender doesn't do it for us:

```
if __name__ == "__main__" :
    register()
#end if
```

124.2. Using The Property

Now we actually need to use the property in our operator's execute routine. We'll use it to negate a scale factor which will be applied to the Z coordinate of the vertices of the tetrahedron:

```
Scale = -1 if context.scene.make_tetrahedron_inverted else 1
```

But only the last vertex has a nonzero Z coordinate, so this is the only one that needs a change in its computation:

```
Vertices = \
[
    mathutils.Vector((0, -1 / math.sqrt(3),0)),
    mathutils.Vector((0.5, 1 / (2 * math.sqrt(3)), 0)),
    mathutils.Vector((-0.5, 1 / (2 * math.sqrt(3)), 0)),
    mathutils.Vector((0, 0, Scale * math.sqrt(2 / 3))),
]
```

124.3. Put It All Together

To recap all our changes, here is the complete updated script:

```
import math
import bpy
import mathutils

class TetrahedronMakerPanel(bpy.types.Panel) :
    bl_space_type = "VIEW_3D"
    bl_region_type = "TOOLS"
    bl_context = "objectmode"
    bl_label = "Add Tetrahedron"

    def draw(self, context) :
        TheCol = self.layout.column(align = True)
        TheCol.prop(context.scene, "make_tetrahedron_inverted")
        TheCol.operator("mesh.make_tetrahedron", text = "Add
Tetrahedron")
#end draw
```

```

#end TetrahedronMakerPanel

class MakeTetrahedron(bpy.types.Operator) :
    bl_idname = "mesh.make_tetrahedron"
    bl_label = "Add Tetrahedron"
    bl_options = {"UNDO"}

    def invoke(self, context, event) :
        Scale = -1 if context.scene.make_tetrahedron_inverted else 1
        Vertices = \
            [
                mathutils.Vector((0, -1 / math.sqrt(3),0)),
                mathutils.Vector((0.5, 1 / (2 * math.sqrt(3)), 0)),
                mathutils.Vector((-0.5, 1 / (2 * math.sqrt(3)), 0)),
                mathutils.Vector((0, 0, Scale * math.sqrt(2 / 3))),
            ]
        NewMesh = bpy.data.meshes.new("Tetrahedron")
        NewMesh.from_pydata \
            (
                Vertices,
                [],
                [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
            )
        NewMesh.update()
        NewObj = bpy.data.objects.new("Tetrahedron", NewMesh)
        context.scene.objects.link(NewObj)
        return {"FINISHED"}
    #end invoke

#end MakeTetrahedron

def register() :
    bpy.types.Scene.make_tetrahedron_inverted =
    bpy.props.BoolProperty \
        (
            name = "Upside Down",
            description = "Generate the tetrahedron upside down",
            default = False
        )
    #end register

def unregister() :
    del bpy.types.Scene.make_tetrahedron_inverted
    #end unregister

if __name__ == "__main__" :
    register()
#end if

```



As before, execute the script with  + . Check the Tool Shelf in the 3D View, and your panel should now look like this:



Figure 910

Try executing it with and without the checkbox checked, and you should end up with two tetrahedra pointing in opposite directions.

125. Separately Installable Addon (version 2.5)

Embedding a script inside a Blender document is useful for some special purposes, but often you want to be able to reuse an addon across any number of Blender documents, and perhaps distribute it to others for use with their Blender projects. To do this, you will need to save the script as a text file that can be copied into your Blender user preferences. It will also need to have some additional information inserted.

125.1. The Addon Info Dictionary

In order for the addon to be installable by Blender, it will need to define a global called `bl_info`, the value of which is a dictionary. This contains various information about compatibility with versions of Blender, and also descriptive information to be shown to the user as they browse through the list of installable addons.

Here's what the info should look like:

```
bl_info = \
{
    "name" : "Tetrahedron Maker",
    "author" : "J Random Hacker <jrhacker@example.com>",
    "version" : (1, 0, 0),
    "blender" : (2, 5, 6),
    "api" : 32411,
    "location" : "View 3D > Edit Mode > Tool Shelf",
    "description" :
        "Generate a tetrahedron mesh",
    "warning" : "",
    "wiki_url" : "",
    "tracker_url" : "",
    "category" : "Add Mesh",
}
```

Most of the fields are informational. The “category” value must be one of the predefined ones that you will see listed in Blender’s User Preferences window when you switch to the “Add-Ons” tab; the addon display is sorted by category and by name within category, and the user can list scripts in all categories or in just one category.

The “version” field indicates the version of your script, and can be a tuple of any number of integers; the numbers are shown to the user joined with decimal points. The “blender” and “api” fields indicate compatibility with versions of Blender; I copied the value 32411 from early example scripts that came with Blender 2.54, and it still seems to work with current 2.56 builds; it may or may not need changing as Blender 2.5x approaches release—watch this space.

125.2. Making The Addon Installable

The contents of your addon script are now complete. In the Text Editor, find the “Save As” option under the “Text” menu, and choose a filename for saving your script—perhaps call it TetrahedronMaker.py.

That’s it. This saved text file is your complete addon, ready for installation into your Blender user settings, or distribution to others for installation in their settings.

125.3. Installing The Addon

To install the addon into your Blender user settings, go to the User Preferences, and bring up the Add-Ons tab. (It doesn’t matter what document you might have open at this stage.) In the window header, click the “Install Add-On...” button, and find the TetrahedronMaker.py file you previously saved; select that, and click the “Install Add-On” button in the file selector. This copies the script file into your personal addons directory, which in Unix/Linux is `~/blender/2.5x/scripts/addons/`. You can also directly copy your script file into this directory, instead of using the “Install Add-On...” button. To uninstall an addon, delete the script from this directory, together with its `.pyc` (compiled) version.

125.4. Using The Installed Addon

To make use of the installed addon, open a new Blender document. Go to the User Preferences window, and bring up the Add-Ons tab. Browse through the list of addons (feel free to restrict the display to the “Add Mesh” category as specified above) until you find your Tetrahedron Maker. You can click the white triangle to the left of the name to expand the list item display to show all the details:

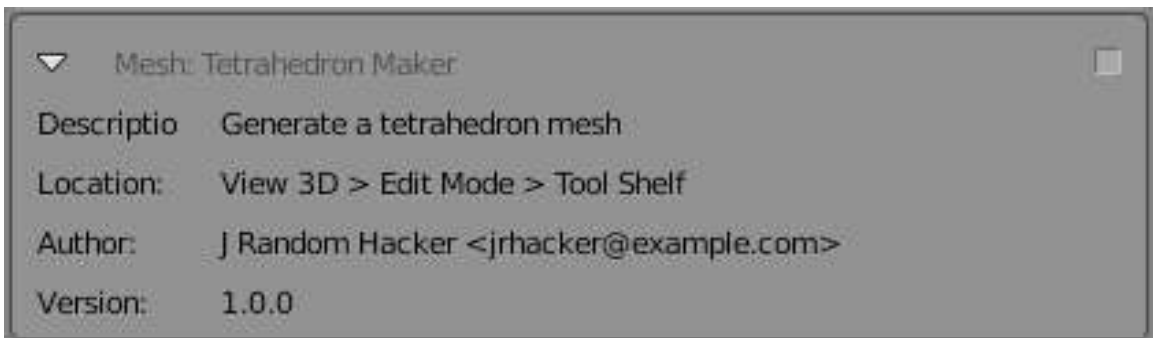


Figure 911

Click the checkbox at the top right of the item display to enable the addon for this document.

Now switch to the 3D View, bring up the Tool Shelf if it’s not already visible, and you should see that familiar panel appear:





Figure 912

You know what to do next...

126. Object, Action, Settings (version 2.5)

Now that we've covered the basics of writing a working addon and making it installable, let's add some refinements to its functionality.

To start with, users normally add new objects via the Add menu which pops up when pressing  + , rather than via a custom panel. Can our script add a new "Tetrahedron" item to that menu? Yes it can.

Also, our script currently requires us to adjust its setting (the "Upside-Down" checkbox) *before* it performs its action. In user-interface parlance, this is the ordering "Object→Setting→Action": select an object, specify the settings for the action, then perform the action. Whereas the preferred order in Blender 2.5x is "Object→Action→Setting": which means the action is performed with some initial settings, which the user is then free to modify while observing their effect. This gives a much smoother workflow, rather than the user having to guess what the effects will be before applying them, and then undoing and trying again if they guessed wrong.

(Of course, since our example script creates a new object rather than modifying an existing one, the initial "Object" step is not relevant here. But it is to other operators.)

So the modifications we need to make to our script are:

- Get rid of the existing Panel subclass.
- Our register function will add a new entry to the Add menu (specifically, the mesh-add submenu) to invoke our Operator subclass.
- When our operator is invoked, a panel will appear, with all the settings controls we previously had, except for the "Add" button. The user can play with these settings, and observe the effect on the newly-created object immediately.

The nice thing is, Blender does most of the hard work for us, so we only need to edit a few lines of code to achieve all the above!

126.1. Adding To The Add Menu

This needs to be done in two steps. First we need a function which will be invoked when Blender creates its Add menu: this will add an entry that invokes our custom operator by its name. Specifically, we will add our custom item to the "Mesh" submenu of the "Add" menu. We can also assign an icon to the menu item; here I'm using the generic plugin icon:

```
def add_to_menu(self, context) :
    self.layout.operator("mesh.make_tetrahedron", icon = "PLUGIN")
#end add_to_menu
```

Having done that, we change our register function to add our `add_to_menu` item to a list that Blender uses to create its Add→Mesh menu:

```
def register() :  
    bpy.types.INFO_MT_mesh_add.append(add_to_menu)  
#end register
```

Note this `bpy.types.INFO_MT_mesh_add` object is not currently mentioned in the API documentation; I found it by examining the scripts that come bundled with Blender.

And of course we should clean up ourselves, so the `unregister` function needs to remove the item we added:

```
def unregister() :  
    bpy.types.INFO_MT_mesh_add.remove(add_to_menu)  
#end unregister
```

126.2. Fixing Up The UI

Note that previously, our `register` and `unregister` functions were attaching a custom property to Blender's `Scene` class to hold our setting value. That code is now gone, but we still need the property. Instead, it will now be attached directly to our `MakeTetrahedron` class.

Also, remember we said we were getting rid of the `TetrahedronMakerPanel` class? In fact we keep the `draw` method from that, and move it to our `MakeTetrahedron` operator, only getting rid of the last button in the panel for invoking the operator. This is because by the time the panel is being drawn, the operator has already been invoked.

We also need to tell Blender that we are conforming to the Object→Settings→Action convention, by adding the "REGISTER" item to our `bl_options`.

So the header of our operator class now looks like this:

```
class MakeTetrahedron(bpy.types.Operator) :  
    bl_idname = "mesh.make_tetrahedron"  
    bl_label = "Tetrahedron"  
    bl_options = {"REGISTER", "UNDO"}  
  
    inverted = bpy.props.BoolProperty \  
    (  
        name = "Upside Down",  
        description = "Generate the tetrahedron upside down",  
        default = False  
    )  
  
    def draw(self, context) :  
        TheCol = self.layout.column(align = True)  
        TheCol.prop(self, "inverted")  
#end draw
```

Previously our custom property was called `make_tetrahedron_inverted`, but here I just call it `inverted`, because after all it is being attached to our own class, so there should be no worry about name clashes.

Note that I also changed the `bl_label` text, taking out the word "Add". This is redundant, because our label will be appearing in a menu that is already titled "Add".

126.3. invoke Versus execute

The last thing to do is rearrange the code for actually creating the tetrahedron object. Previously we had it in a method called `invoke`, and that method will still be called when our operator is selected from the `Add→Mesh` menu. Then our `draw` method will be called, so our panel will also appear. But then, if the user makes adjustments to the controls in our panel, Blender will invoke a different operator method, called `execute`.

To the user, it looks like they are making adjustments to an already-created object. But in fact our `execute` method can do exactly the same thing as `invoke`, namely create a new object every time it's called! Blender will take care of getting rid of previously-created objects, so the user won't know the difference.

Nice, isn't it? But in order for this to work, our code needs to do one thing more: ensure that our newly-created object is the only selected, active object. So the following is added after `NewObj` has been created and linked into the scene:

```
bpy.ops.object.select_all(action = "DESELECT")
NewObj.select = True
context.scene.objects.active = NewObj
```

126.4. Put It All Together

Note that the object-creation code has been moved into a common routine that I have called `action_common`. This can then be called from both the `invoke` and `execute` methods.

```
import math
import bpy
import mathutils

class MakeTetrahedron(bpy.types.Operator) :
    bl_idname = "mesh.make_tetrahedron"
    bl_label = "Tetrahedron"
    bl_options = {"REGISTER", "UNDO"}

    inverted = bpy.props.BoolProperty \
    (
        name = "Upside Down",
        description = "Generate the tetrahedron upside down",
        default = False
    )

    def draw(self, context) :
        TheCol = self.layout.column(align = True)
        TheCol.prop(self, "inverted")
    #end draw

    def action_common(self, context) :
        Scale = -1 if self.inverted else 1
        Vertices = \
        [
            mathutils.Vector((0, -1 / math.sqrt(3), 0)),
            mathutils.Vector((0.5, 1 / (2 * math.sqrt(3)), 0)),
            mathutils.Vector((-0.5, 1 / (2 * math.sqrt(3)), 0)),
            mathutils.Vector((0, 0, Scale * math.sqrt(2 / 3))),
        ]
        NewMesh = bpy.data.meshes.new("Tetrahedron")
```

```
NewMesh.from_pydata \
(
    Vertices,
    [],
    [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
)
NewMesh.update()
NewObj = bpy.data.objects.new("Tetrahedron", NewMesh)
context.scene.objects.link(NewObj)
bpy.ops.object.select_all(action = "DESELECT")
NewObj.select = True
context.scene.objects.active = NewObj
#end action_common

def execute(self, context) :
    self.action_common(context)
    return {"FINISHED"}
#end execute

def invoke(self, context, event) :
    self.action_common(context)
    return {"FINISHED"}
#end invoke

#end MakeTetrahedron

def add_to_menu(self, context) :
    self.layout.operator("mesh.make_tetrahedron", icon = "PLUGIN")
#end add_to_menu

def register() :
    bpy.types.INFO_MT_mesh_add.append(add_to_menu)
#end register

def unregister() :
    bpy.types.INFO_MT_mesh_add.remove(add_to_menu)
#end unregister

if __name__ == "__main__" :
    register()
#end if
```

126.5. Undocumented Blender

That `INFO_MT_mesh_add` object is one of a bunch of undocumented things lurking in the `bpy.types` module. They all have names of the form *prefix_type_restofname*, where *prefix* is an all-upercase mnemonic for the category of object (mostly a window type, e.g. INFO for the Info window, which is where the main menu bar appears, though there are also names for major object categories like MESH and LAMP), and *type* indicates the class of object: HT for a window header UI object (`bpy.types.Header`), MT for a menu (`bpy.types.Menu`), OT for an operator, and PT for a panel (`bpy.types.Panel`). The OT ones seem to correspond directly to objects in `bpy.ops`, but at a lower level; it's probably easier just to use the official `bpy.ops` objects. As for the HT, MT and PT objects, these all have `append`, `prepend` and `remove` methods that you can use to customize them: `append` adds a widget at the end (right or bottom), while `prepend` puts it at the front (left or top).

Since they aren't (yet) mentioned in the official Blender documentation, here is a list of all the ones I've been able to find:

Operator Category	Header	Menu	Panel
Bone Properties			BONE_PT_constraints, BONE_PT_context_bone, BONE_PT_custom_props, BONE_PT_deform, BONE_PT_display, BONE_PT_inverse_kinematics, BONE_PT_relations, BONE_PT_transform, BONE_PT_transform_locks
Cloth Properties		CLOTH_MT_presets	
Console	CONSOLE_HT_header	CONSOLE_MT_console, CONSOLE_MT_language	

Operator Category	Header	Menu	Panel
Data			DATA_PT_active_- spline, DATA_- PT_area, DATA_- PT_bone_groups, DATA_PT_camera, DATA_PT_camera_- display, DATA_- PT_context_arm, DATA_PT_context_- camera, DATA_- PT_context_curve, DATA_PT_context_- lamp, DATA_PT_- context_lattice, DATA_PT_context_- mesh, DATA_PT_- context_metaball, DATA_PT_custom_- props_arm, DATA_- PT_custom_props_- camera, DATA_- PT_custom_props_- curve, DATA_PT_- custom_props_- lamp, DATA_PT_- custom_props_- lattice, DATA_PT_- custom_props_- mesh, DATA_- PT_custom_- props_metaball, DATA_PT_display, DATA_PT_empty, DATA_PT_falloff_- curve, DATA_- PT_font, DATA_- PT_geometry_- curve, DATA_- PT_ghost, DATA_- PT_iksolver_- itasc, DATA_PT_- lamp, DATA_PT_- lattice, DATA_PT_- metaball, DATA_- PT_metaball_- element, DATA_- PT_modifiers, DATA_PT_motion_- paths, DATA_PT_- normals, DATA_- PT_paragraph, DATA_PT_- pathanim, DATA_- PT_preview, DATA_PT_settings, DATA_PT_

Operator Category	Header	Menu	Panel
Dopesheet	DOPE SHEET_ HT_header	DOPE SHEET_ MT_channel, DOPE SHEET_ MT_gpencil_ channel, DOPE SHEET_ MT_gpencil_frame, DOPE SHEET_ MT_key, DOPE SHEET_ MT_key_transform, DOPE SHEET_ MT_marker, DOPE SHEET_ MT_select, DOPE SHEET_ MT_view	
Filebrowser	FILEBROWSER_ HT_header		
Graph Editor	GRAPH_HT_ header	GRAPH_MT_ channel, GRAPH_ MT_key, GRAPH_ MT_key_transform, GRAPH_MT_ marker, GRAPH_ MT_select, GRAPH_MT_view	

Operator Category	Header	Menu	Panel
UV/Image Editor	IMAGE_HT_- header	IMAGE_MT_- image, IMAGE_- MT_select, IMAGE_MT_uvsv, IMAGE_MT_uvsv_- mirror, IMAGE_- MT_uvsv_select_- mode, IMAGE_- MT_uvsv_showhide, IMAGE_MT_uvsv_- snap, IMAGE_- MT_uvsv_transform, IMAGE_MT_- uvsv_weldalign, IMAGE_MT_view	IMAGE_PT_- game_properties, IMAGE_PT_- image_properties, IMAGE_PT_- paint, IMAGE_- PT_paint_curve, IMAGE_PT_paint_- stroke, IMAGE_- PT_sample_line, IMAGE_PT_- scope_sample, IMAGE_PT_tools_- brush_texture, IMAGE_PT_- tools_brush_tool, IMAGE_PT_- view_histogram, IMAGE_PT_- view_properties, IMAGE_PT_- view_vectorscope, IMAGE_PT_view_- waveform
Info Window	INFO_HT_header	INFO_MT_- add, INFO_MT_- armature_add, INFO_MT_curve_- add, INFO_MT_- edit_curve_add, INFO_MT_file, INFO_MT_file_- export, INFO_MT_- file_external_data, INFO_MT_file_- import, INFO_MT_- game, INFO_MT_- help, INFO_MT_- mesh_add, INFO_- MT_render, INFO_- MT_report, INFO_- MT_surface_add	
Lamp Properties		LAMP_MT_- sunsky_presets	

Operator Category	Header	Menu	Panel
Logic Editor	LOGIC_HT_header	LOGIC_MT_- logicbricks_add, LOGIC_MT_view	LOGIC_PT_- properties
Material Properties		MATERIAL_- MT_specials, MATERIAL_MT_- sss_presets	MATERIAL_PT_- context_material, MATERIAL_PT_- custom_props, MATERIAL_- PT_diffuse, MATERIAL_- PT_flare, MATERIAL_PT_- halo, MATERIAL_- PT_mirror, MATERIAL_- PT_options, MATERIAL_- PT_physics, MATERIAL_- PT_preview, MATERIAL_- PT_shading, MATERIAL_- PT_shadow, MATERIAL_- PT_specular, MATERIAL_PT_- sss, MATERIAL_- PT_strand, MATERIAL_- PT_transp, MATERIAL_- PT_transp_game, MATERIAL_PT_- volume_density, MATERIAL_PT_- volume_integration, MATERIAL_PT_- volume_lighting, MATERIAL_PT_- volume_options, MATERIAL_PT_- volume_shading, MATERIAL_PT_- volume_transp

Operator Category	Header	Menu	Panel
Mesh Properties		MESH_MT_- shape_key_specials, MESH_MT_- vertex_group_- specials	
NLA Editor	NLA_HT_header	NLA_MT_add, NLA_MT_edit, NLA_MT_edit_- transform, NLA_- MT_marker, NLA_- MT_select, NLA_- MT_view	
Node Editor	NODE_HT_header	NODE_MT_node, NODE_MT_select, NODE_MT_view	
Object Properties			OBJECT_- PT_animation, OBJECT_PT_- constraints, OBJECT_PT_- context_object, OBJECT_PT_- custom_props, OBJECT_PT_- delta_transform, OBJECT_PT_- display, OBJECT_- PT_duplication, OBJECT_PT_- groups, OBJECT_- PT_motion_paths, OBJECT_PT_- relations, OBJECT_- PT_transform, OBJECT_PT_- transform_locks
Outliner	OUTLINER_HT_- header	OUTLINER_MT_- edit_datablocks, OUTLINER_- MT_search, OUTLINER_MT_- view	

Operator Category	Header	Menu	Panel
Particle Properties			PARTICLE_- PT_boidbrain, PARTICLE_PT_- cache, PARTICLE_- PT_children, PARTICLE_PT_- context_particles, PARTICLE_PT_- custom_props, PARTICLE_PT_- draw, PARTICLE_- PT_emission, PARTICLE_PT_- field_weights, PARTICLE_- PT_force_fields, PARTICLE_PT_- hair_dynamics, PARTICLE_- PT_physics, PARTICLE_- PT_render, PARTICLE_- PT_rotation, PARTICLE_- PT_velocity, PARTICLE_PT_- vertexgroups

Operator Category	Header	Menu	Panel
Physics Properties			PHYSICS_PT_- cloth, PHYSICS_- PT_cloth_cache, PHYSICS_PT_- cloth_collision, PHYSICS_PT_- cloth_field_weights, PHYSICS_PT_- cloth_stiffness, PHYSICS_- PT_collision, PHYSICS_PT_- domain_boundary, PHYSICS_PT_- domain_gravity, PHYSICS_PT_- domain_particles, PHYSICS_PT_field, PHYSICS_PT_fluid, PHYSICS_PT_- game_collision_- bounds, PHYSICS_- PT_game_physics, PHYSICS_PT_- smoke, PHYSICS_- PT_smoke_- cache, PHYSICS_- PT_smoke_- field_weights, PHYSICS_PT_- smoke_groups, PHYSICS_PT_- smoke_highres, PHYSICS_- PT_softbody, PHYSICS_PT_- softbody_cache, PHYSICS_PT_- softbody_collision, PHYSICS_PT_- softbody_edge, PHYSICS_- PT_softbody_- field_weights, PHYSICS_PT_- softbody_goal, PHYSICS_PT_- softbody_solver

Operator Category	Header	Menu	Panel
Render Properties		RENDER_MT_ffmpeg_presets, RENDER_MT_presets	RENDER_PT_antialiasing, RENDER_PT_bake, RENDER_PT_dimensions, RENDER_PT_encoding, RENDER_PT_game, RENDER_PT_game_performance, RENDER_PT_game_player, RENDER_PT_game_shading, RENDER_PT_game_sound, RENDER_PT_game_stereo, RENDER_PT_layers, RENDER_PT_motion_blur, RENDER_PT_network_job, RENDER_PT_network_job_vcs, RENDER_PT_network_jobs, RENDER_PT_network_master_settings, RENDER_PT_network_output, RENDER_PT_network_settings, RENDER_PT_network_slave_settings, RENDER_PT_network_slaves, RENDER_PT_network_slaves_blacklist, RENDER_PT_output, RENDER_PT_performance, RENDER_PT_post_processing, RENDER_PT_render, RENDER_PT_shading, RENDER_PT_stamp

Operator Category	Header	Menu	Panel
Scene Properties			SCENE_PT_- custom_props, SCENE_PT_- keying_set_- paths, SCENE_- PT_keying_sets, SCENE_PT_- physics, SCENE_- PT_scene, SCENE_- PT_simplify, SCENE_PT_unit
Sequencer Window	SEQUENCER_- HT_header	SEQUENCER_- MT_add, SEQUENCER_- MT_add_effect, SEQUENCER_- MT_marker, SEQUENCER_- MT_select, SEQUENCER_- MT_strip, SEQUENCER_- MT_view, SEQUENCER_- MT_view_toggle	SEQUENCER_- PT_edit, SEQUENCER_- PT_effect, SEQUENCER_- PT_filter, SEQUENCER_- PT_input, SEQUENCER_- PT_preview, SEQUENCER_- PT_proxy, SEQUENCER_- PT_scene, SEQUENCER_- PT_sound, SEQUENCER_- PT_view

Operator Category	Header	Menu	Panel
Texture Properties		TEXTURE_MT_- envmap_specials, TEXTURE_MT_- specials	TEXTURE_PT_- blend, TEXTURE_- PT_clouds, TEXTURE_PT_- colors, TEXTURE_- PT_context_texture, TEXTURE_PT_- custom_props, TEXTURE_PT_- distortednoise, TEXTURE_- PT_envmap, TEXTURE_PT_- envmap_sampling, TEXTURE_PT_- image, TEXTURE_- PT_image_mapping, TEXTURE_PT_- image_sampling, TEXTURE_- PT_influence, TEXTURE_PT_- magic, TEXTURE_- PT_mapping, TEXTURE_- PT_marble, TEXTURE_- PT_musgrave, TEXTURE_- PT_pointdensity, TEXTURE_PT_- pointdensity_- turbulence, TEXTURE_- PT_preview, TEXTURE_PT_- stucci, TEXTURE_- PT_voronoi, TEXTURE_- PT_voxeldata, TEXTURE_PT_- wood

Operator Category	Header	Menu	Panel
Text Editor	TEXT_HT_header	TEXT_MT_edit, TEXT_MT_edit_ markers, TEXT_ MT_edit_select, TEXT_MT_edit_ to3d, TEXT_MT_ format, TEXT_ MT_templates, TEXT_MT_text, TEXT_MT_toolbox, TEXT_MT_view	TEXT_PT_ find, TEXT_PT_ properties
Timeline	TIME_HT_header	TIME_MT_ autokey, TIME_ MT_cache, TIME_ MT_frame, TIME_ MT_playback, TIME_MT_view	
User Preferences	USERPREF_HT_ header	USERPREF_MT_ addons_dev_guides, USERPREF_MT_ interaction_presets, USERPREF_ MT_keyconfigs, USERPREF_MT_ splash	USERPREF_ PT_addons, USERPREF_PT_ edit, USERPREF_ PT_file, USERPREF_PT_ input, USERPREF_ PT_interface, USERPREF_ PT_system, USERPREF_PT_ tabs, USERPREF_ PT_theme

Operator Category	Header	Menu	Panel
3D View	VIEW3D_HT_- header	VIEW3D_MT_- armature_specials, VIEW3D_MT_- edit_armature, VIEW3D_MT_- edit_armature_- parent, VIEW3D_- MT_edit_armature_- roll, VIEW3D_- MT_edit_curve, VIEW3D_- MT_edit_- curve_ctrlpoints, VIEW3D_- MT_edit_- curve_segments, VIEW3D_- MT_edit_- curve_showhide, VIEW3D_MT_- edit_curve_- specials, VIEW3D_- MT_edit_font, VIEW3D_MT_- edit_lattice, VIEW3D_- MT_edit_mesh, VIEW3D_MT_- edit_mesh_edges, VIEW3D_MT_- edit_mesh_extrude, VIEW3D_MT_- edit_mesh_faces, VIEW3D_MT_- edit_mesh_normals, VIEW3D_MT_- edit_mesh_select_- mode, VIEW3D_- MT_edit_- mesh_showhide, VIEW3D_MT_- edit_mesh_specials, VIEW3D_MT_- edit_mesh_- vertices, VIEW3D_- MT_edit_meta, VIEW3D_- MT_edit_- meta_showhide, VIEW3D_MT_- edit_surface, VIEW3D_MT_- edit_text_chars, VIEW3D_MT_- hook, VIEW3D_-	VIEW3D_PT_- background_image, VIEW3D_PT_- context_properties, VIEW3D_PT_etch_- a_ton, VIEW3D_- PT_sculpt_options, VIEW3D_PT_- sculpt_symmetry, VIEW3D_PT_- tools_armatureedit, VIEW3D_PT_- tools_armatureedit_- options, VIEW3D_- PT_tools_brush, VIEW3D_PT_- tools_brush_- appearance, VIEW3D_PT_- tools_brush_curve, VIEW3D_PT_- tools_brush_stroke, VIEW3D_PT_- tools_brush_texture, VIEW3D_PT_- tools_brush_tool, VIEW3D_PT_- tools_curveedit, VIEW3D_PT_- tools_latticeedit, VIEW3D_PT_- tools_mballedit, VIEW3D_PT_- tools_meshedit, VIEW3D_PT_- tools_meshedit_- options, VIEW3D_- PT_tools_- objectmode, VIEW3D_PT_- tools_particlemode, VIEW3D_PT_- tools_posemode, VIEW3D_PT_- tools_posemode_- options, VIEW3D_- PT_tools_- projectpaint, VIEW3D_PT_- tools_surfaceedit, VIEW3D_PT_- tools_textedit, VIEW3D_PT_- tools_vertexpaint, VIEW3D_PT_-

Operator Category	Header	Menu	Panel
Window Manager Properties		WM_MT_- operator_presets	
World Properties			WORLD_PT_- ambient_occlusion, WORLD_PT_- context_world, WORLD_PT_- custom_props, WORLD_PT_- environment_- lighting, WORLD_- PT_game_context_- world, WORLD_- PT_game_mist, WORLD_PT_- game_physics, WORLD_PT_- game_world, WORLD_PT_- gather, WORLD_- PT_indirect_- lighting, WORLD_- PT_mist, WORLD_- PT_preview, WORLD_PT_stars, WORLD_PT_world

127. Procedural object creation

Information

Applicable Blender version: 2.49

127.1. Text Blocks

A Blender document can contain *text blocks*, which are not the same as *text objects* in a 3D scene (though the former can be converted to the latter). Besides generating text objects, a text block can serve any purpose you like; for example, use it to pass workflow instructions to a colleague along with the document; display a copyright or help message in the initial layout that a user sees on opening the document; or hold a Python script that can be run by the user to perform some useful action related to the document.

Text blocks are edited in the Text Editor window. The Text Editor also provides commands to load the contents of a text block from an external file, or save it to an external file. And also execute the text as a Python script.

127.2. Your First Script

Open a new, default Blender document. Split the 3D View in two vertically. Change the type of one side to a Text Editor window. In the header, you will see a small popup menu showing just a double-headed arrow; click on this, and it should show three items: “Text”, “ADD NEW” and “OPEN NEW”. “ADD NEW” creates a new, empty text block, while “OPEN NEW” creates a new text block by reading its contents from an external file. But “Text” is already the name of a default empty text block, so just use that; as soon as you select it, you should see a red insertion cursor appear at the top left, indicating that you can start typing.

Unlike the Interactive Python Console, nothing is automatically imported for you. So as in any other Python script, you need to mention every module you want to access.

Let us write a script that inserts a new mesh primitive into a Blender document, namely a tetrahedron. First we need to create a new mesh datablock, which we will name “Tetrahedron”:

```
NewMesh = Blender.Mesh.New("Tetrahedron")
```

Then we need to define the coordinates of the vertices; for a tetrahedron with edges of length 1 Blender unit, suitable values are $(0, -\frac{1}{\sqrt{3}}, 0)$, $(\frac{1}{2}, \frac{1}{2\sqrt{3}}, 0)$, $(-\frac{1}{2}, \frac{1}{2\sqrt{3}}, 0)$ and $(0, 0, \sqrt{\frac{2}{3}})$. Or in Python:

```
NewMesh.verts.extend \  
(
```

```
[
    (0, -1 / math.sqrt(3), 0),
    (0.5, 1 / (2 * math.sqrt(3)), 0),
    (-0.5, 1 / (2 * math.sqrt(3)), 0),
    (0, 0, math.sqrt(2 / 3)),
]
```

We also need to define the faces of the object; each face is defined by listing a sequence of indexes into the above array of vertices (you don't need to bother about defining the edges; Blender will deduce them from the faces):

```
NewMesh.faces.extend \
(
    [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
)
```

That suffices for the mesh, now we create an actual object datablock that will appear in the scene (which we also name "Tetrahedron"):

```
TheObj = Blender.Object.New("Mesh", "Tetrahedron")
```

Link it to the mesh we just made:

```
TheObj.link(NewMesh)
```

And to make the object appear in the scene, it has to be linked to it:

```
TheScene = Blender.Scene.GetCurrent()
TheScene.link(TheObj)
```

And finally, tell Blender that the scene has changed:

```
TheScene.update()
```

and to redraw the 3D view to show the updated scene:

```
Blender.Window.Redraw()
```

127.3. Put It All Together

Your complete script should look like this, including the imports of the referenced math and Blender modules; note also the use of the directive to ensure that the "/" operator always returns a real, not an integer, result; this is good practice with Python 2.x, since it becomes mandatory behaviour beginning with Python 3.0.

```
from __future__ import division
import math
import Blender

NewMesh = Blender.Mesh.New("Tetrahedron")
NewMesh.verts.extend \
(
    [
        (0, -1 / math.sqrt(3), 0),
        (0.5, 1 / (2 * math.sqrt(3)), 0),
```

```

        (-0.5, 1 / (2 * math.sqrt(3)), 0),
        (0, 0, math.sqrt(2 / 3)),
    ]
)
NewMesh.faces.extend \
(
    [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
)
TheObj = Blender.Object.New("Mesh", "Tetrahedron")
TheObj.link(NewMesh)
TheScene = Blender.Scene.GetCurrent()
TheScene.link(TheObj)
TheScene.update()
Blender.Window.Redraw()

```

In your 3D View, get rid of any default cube to avoid it obscuring things. Now come back to the Text Editor window showing the above script, and press **ALT** + **P** to execute it; you should see your new tetrahedron appear in the 3D view!

127.4. If You Hit An Error

If there is any error running the script, Blender will display a cryptic message to let you know. For example, the following simple one-line script

```
raise RuntimeError("Uh-oh")
```

displays this message:



Figure 913

To get more details you will have to look in Standard Error; on Linux/Unix systems, the message will appear in the terminal session if you invoked Blender from the command line; otherwise it will be appended to your `~/.xsessionerrors` file if you launched Blender from a GUI. On Windows the message appears in the console window. By hunting around in the appropriate place, you should be able to find the full Python traceback:

```

Traceback (most recent call last):
  File "Text.001", line 1, in <module>
RuntimeError: Uh-oh

```


128. External scripts

Information

Applicable Blender version: 2.49

Embedding a script inside a Blender document is useful for some special purposes, but often you want to be able to reuse it across any number of Blender documents, and perhaps distribute it to others for use with their Blender projects. To do this, you will need to save the script as a text file that can be copied into your Blender user preferences. It will also need to have some additional information inserted.

128.1. The Script Header

In order for your script to be installable, it needs to have the following as its first line:

```
#!/BPY
```

Following this, add a multiline Python string to define a name and other information for the script to appear in the Scripts menu:

```
"""
Name: 'Add Tetrahedron'
Blender: 249
Group: 'Add'
Tooltip: 'Creates a tetrahedron object'
"""
```

The “Blender:” line specifies the minimum required version of Blender that the script is compatible with; here we specify “249” to indicate that it wants Blender version 2.49. The “Group:” indicates which of the submenus of the Scripts menu that the script will appear in: it must match one of the existing submenu names, otherwise the script will go into the “Misc” submenu. The “Name:” is the name of the menu item that will be added for invoking this script, and the “Tooltip:” will appear when the user pauses with the mouse over that menu item.

128.2. Put It All Together

So now your complete script looks like this:

```
#!/BPY
"""
Name: 'Add Tetrahedron'
Blender: 249
Group: 'Add'
```



```
Tooltip: 'Creates a tetrahedron object'
"""

from __future__ import division
import math
import Blender

NewMesh = Blender.Mesh.New("Tetrahedron")
NewMesh.verts.extend \
(
    [
        (0, -1 / math.sqrt(3), 0),
        (0.5, 1 / (2 * math.sqrt(3)), 0),
        (-0.5, 1 / (2 * math.sqrt(3)), 0),
        (0, 0, math.sqrt(2 / 3)),
    ]
)
NewMesh.faces.extend \
(
    [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
)
TheObj = Blender.Object.New("Mesh", "Tetrahedron")
TheObj.link(NewMesh)
TheScene = Blender.Scene.GetCurrent()
TheScene.link(TheObj)
TheScene.update()
Blender.Window.Redraw()
```

128.3. Making The Script Installable

The contents of your script are now complete. In the Text Editor, find the “Save As” option under the “Text” menu, and choose a filename for saving your script—perhaps call it TetrahedronMaker.py.

That’s it. This saved text file is now ready for installation into your Blender user settings, or distribution to others for installation in their settings.

128.4. Installing The Script

To install the script, simply copy the file you previously saved into the user-scripts directory within your Blender user settings.

For different operating systems this is:

- Linux/Unix: `~/blender/scripts`
- Windows XP: `C:\Program Files\Blender Foundation\Blender\blender\scripts`
- Windows XP (alt): `C:\Documents and Settings\USERNAME\Application Data\Blender Foundation\Blender\blender\scripts`
- Windows Vista: `C:\Users\USERNAME\AppData\Roaming\Blender Foundation\Blender\blender\scripts`
- Mac OS X:
 - Under Mac OSX the path is actually hidden in the blender.app so to know the path you would have to know that the script folder is actually hidden in the blender.app itself. Assuming that Blender is in the applications directory the path would be `"/Applications/blender/blender.app/Contents/MacOS/blender/scripts"` If you try to open the .app contents

from the finder you will notice that `.blender` section of the path is not visible, while blender will still be able to navigate to this folder.

- Right-click (or ctrl-click) the file "blender", and select "Show Package Contents" in the popup-menu. It will display all the hidden files under blender's folder, and select "scripts" folder inside it.
- To see this folder from the OSX terminal use the `ls -a` command (lists all folders/files even hidden) in the MacOS folder of the listed path. It is probably a good idea to create an alias to the scripts folder in the `/Applications/blender-2.37a-OSX-10.3-powerpc` folder so that scripts can be easily manipulated through the finder. I know that its confusing that Blender should have its script folder buried inside the app but it is necessary to keep the app portable and not require an install.
- A safer approach than the one above consists in keeping your scripts somewhere in your home folder: with this scheme, there is no risk of deleting your scripts when you upgrade your blender application, as they are not contained within its folder. A method that follows this principle is as follows: create a folder that will contain your scripts (or some of them) inside your own home directory; then, instead of putting your files directly in the `.../.blender/scripts/` folder discussed above, simply add a *link* to your script directory in the `.../.blender/scripts/` folder (for instance with the "ln -s" Unix command, or by doing "open /Applications/blender-2.37a-OSX-10.3-powerpc/blender.app/Contents/MacOS/.blender/scripts/" [adapted to your version of blender] and then creating a link through the Finder, with File->Make Alias). Blender will now find all the scripts that you put in your home directory: it will follow the link you created in its `.../.blender/scripts/` folder and go to the corresponding folder in your own directory, and find all the python scripts you put there.

128.5. Using The Installed Script

To make use of the installed script, open a new Blender document. Get rid of the default cube to avoid it obscuring things. Split the 3D View into two horizontally, and change one of the two into a Scripts Window. Go to the "Scripts" menu, and see if your "Add Tetrahedron" item appears in the "Add" submenu; if it doesn't, select "Update Menus" (the last item in the "Scripts" menu) and have another look. Select your new menu item, and watch your tetrahedron object appear in the 3D View!

129. Creating a GUI for your script

i Information

Applicable Blender version: 2.49

Often, your script will need to get information from the user, such as parameter settings, before performing its action. You can do this in two different ways: either present a modal popup that the user has to deal with before doing anything else, or alternatively you can take over the entire Scripts Window with a custom display that the user can freely switch to and from while doing other things.

129.1. A Modal GUI

The simplest way to add a GUI to your script is to pop up a simple menu with some items to choose from. Let's add an option to our Tetrahedron Maker script to generate the tetrahedron upside-down. Displaying a menu is as simple as calling `Blender.Draw.PupMenu` and passing it a string containing the menu items separated by “|” characters. Put a “%t” at the end of the item that defines the menu title:

```
Orient = Blender.Draw.PupMenu("Add Tetrahedron%t|Right Way Up|Upside  
Down")
```

The value returned from the call will be the index of the selected menu item (starting from 1, ignoring the menu title), or -1 to indicate that no selection was made.

When you execute the above line, it displays this:



Figure 914

We will use the menu selection to negate a scale factor which will be applied to the Z coordinate of the vertices of the tetrahedron. But only the last vertex has a nonzero Z coordinate, so this is the only one that needs a change in its computation. So the body of our script now looks like this:

```
Orient = Blender.Draw.PupMenu("Add Tetrahedron&t|Right Way Up|Upside
Down")
if Orient > 0 :
    Scale = -1 if Orient > 1 else 1
    NewMesh = Blender.Mesh.New("Tetrahedron")
    NewMesh.verts.extend \
    (
        [
            (0, -1 / math.sqrt(3),0),
            (0.5, 1 / (2 * math.sqrt(3)), 0),
            (-0.5, 1 / (2 * math.sqrt(3)), 0),
            (0, 0, Scale * math.sqrt(2 / 3)),
        ]
    )
    NewMesh.faces.extend \
    (
        [[0, 1, 2], [0, 1, 3], [1, 2, 3], [2, 0, 3]]
    )
    TheObj = Blender.Object.New("Mesh", "Tetrahedron")
    TheObj.link(NewMesh)
    TheScene = Blender.Scene.GetCurrent()
    TheScene.link(TheObj)
    TheScene.update()
    Blender.Window.Redraw()
#end if
```

129.2. A Modeless GUI

It is very easy to create a GUI for your script, and that way make it easy to change aspects of it for everyone.

The command to create a Graphical User Interface (GUI) is:

```
Blender.Draw.Register(draw,event,button)
```

This command registers the functions:

- draw - to draw the GUI
- event - to action mouse and key presses
- button - to action GUI button presses

However, this command will NOT work by itself !!!. You first need to define these 3 functions.

First we will import Blender's library of built in functions:

```
import Blender
```

Next, we will define the draw function.

```
def draw():
```

Inside this function we will draw the GUI. Here is an example of a drawing function we can use. It will clear the current window.

```
Blender.BGL.glClear(Blender.BGL.GL_COLOR_BUFFER_BIT)
```

And the next command will draw a button. Note that the first number in the command, '1' identifies the button as **button 1**. We will refer to this button later.

```
Blender.Draw.Toggle("Clear origin",1,10,20,100,20,0,"Tooltip")
```

Next, we will define the event function. The code of a key pressed on the keyboard is passed into the function as the variable **evt**.

```
def event(evt,val):
```

Now we will test to see if the escape key is pressed:

```
if evt == Blender.Draw.ESCKEY:
```

If it is pressed, exit the script, and return from the function:

```
Blender.Draw.Exit()
return
```

Next, we will define the button function. This function will perform an action if the button is pressed.

```
def button(evt):
```

Now test the variable **evt** which holds the button number that we previously identified.

```
if evt == 1:
```

If it is pressed, we will move the selected object in the 3d window back to the centre and redraw the screen:

```
Blender.Scene.GetCurrent().getActiveObject().loc = (0,0,0)
Blender.Window.Redraw()
```

Lastly, we can create the Graphical User Interface by typing the command:

```
Blender.Draw.Register(draw,event,button)
```

That's it !!! To enter the script yourself, type the following into the Text Editor window in Blender, and then press **alt p** to execute the script. Here's the entire script. Everything after the hash # is a comment and can be left out.

```
import Blender # This will import the library of blender functions
we will use

def draw(): # Define the draw function (which draws your GUI).
    Blender.BGL.glClear(Blender.BGL.GL_COLOR_BUFFER_BIT) # This
    clears the window
    # Add here drawing commands to draw your GUI, for example:
    Blender.Draw.Toggle("Clear origin",1,10,20,100,20,0,"Tooltip")
    # The line above will draw a toggle button.
    # Note the first number, '1' means this is button number 1

def event(evt,val): # Define mouse and keyboard press events
    if evt == Blender.Draw.ESCKEY: # Example if esc key pressed
        Blender.Draw.Exit() # then exit script
        return # return from the function

def button(evt): # Define what to do if a button is pressed, for
example:
    if evt == 1: # If button '1' is pressed, set active object to
centre:
        Blender.Scene.GetCurrent().getActiveObject().loc = (0,0,0)
        Blender.Window.Redraw() # This will redraw the 3d window.

# You can now run the Graphical User Interface by typing the command:
```

```
Blender.Draw.Register(draw,event,button)
# End of script
```

Noob Note: When I tried to run that script it came up with an error: Attribute Error: 'Scene' object has no attribute 'getActiveObject'. I changed "Blender.Scene.GetCurrent().getActiveObject().loc = (0,0,0)" to "Blender.Scene.GetCurrent().**objects.active**.loc = (0,0,0)" and it worked.

130. Export scripts

Information

Applicable Blender version: 2.59

This page is an update of a great tutorial originally built for blender 2.44

Blender is not just useful to create complete animations, but it's also a great modeller. You can build your complete 3D scene in Blender, and then export it to a useful format. In fact, you can use it for much more, for example I was using it as a level editor for a freeware 2D game someone else made. There was a short deadline for the game to be finished, and 2 weeks before that deadline, there still was no level editor for it. It had a custom ASCII level format, consisting of lists of materials, vertices, triangles and objects. So, remembering the Blender Python exporters, I volunteered to write an export script for Blender, so it could be used as level editor. And it worked out very well, Blender can be completely used as level editor for that game now.

In this tutorial we'll learn how to write a simple Python export script for Blender. Without requiring previous Python knowledge, it will explain how to query the objects in your scene, and how to write them to a file. It will also demonstrate the usefulness of export scripts, by showing how you can process the data while exporting, so you can achieve things that would not work by using any other existing format.

So, open Blender, make sure the default scene is loaded, and let's begin..

130.1. Finding out about things in a scene

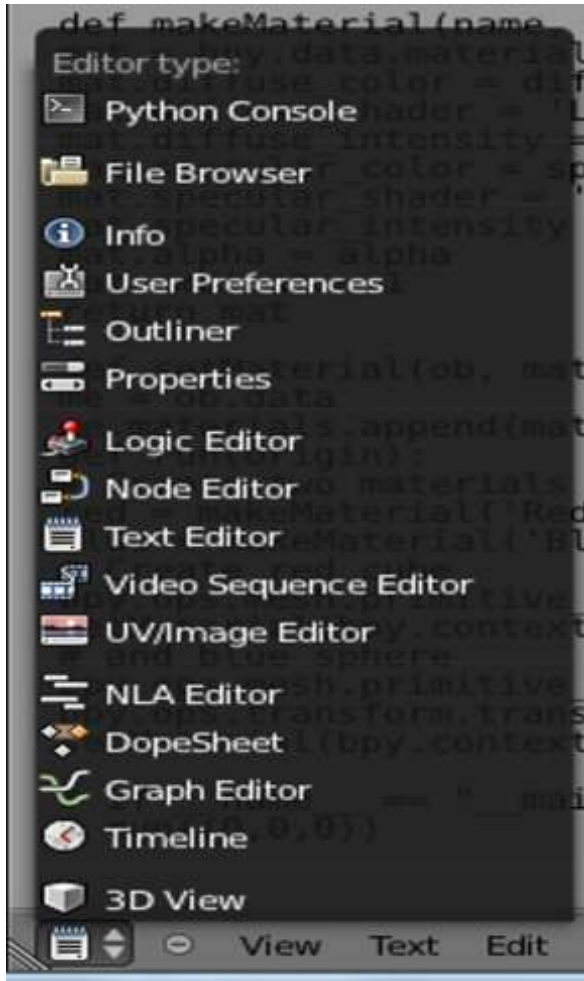


Figure 915 Blender259 EditorSelector

Before we can export something, we must know what to export. One way to get this information is the Outliner window (**SHIFT-F9**). It will list all the things currently known to Blender. Now, we want the same information from a script. Click the editor selection button and from the popup menu choose **Python Console**.

Now, you are ready for the big moment, you are about to execute the first Blender scripting command. Type this and hit **RETURN** (or you could type into the scripts window `import Blender` on the topmost line, then these lines below it, precede all the 'dir(x)' lines with `print` and choose `file->Execute`):

```
list(bpy.data.objects)
```

As a result, you should see this:

```
[bpy.data.objects["Camera"], bpy.data.objects["Cube"],  
bpy.data.objects["Lamp"]]
```

Now, what just happened? The variable given to "list" in "list(bpy.data.objects)" consists of three words, separated by two dots. The dots separate different things. The first, *bpy*, means to use a function from the *bpy* module. *data* is a sub-module of Blender. And finally *objects* is an iterator of *bpy.data*. The *list()* function is used to loop through all data in *bpy.data.objects* and return that as a list of all available objects. In our case, this is a Camera, a Cube, and a Lamp.

To get more information about an object, you can use the object name as a key in *bpy.data.objects*, and assign it to a variable, like this:

```
camera = bpy.data.objects["Camera"]  
cube = bpy.data.objects["Cube"]  
lamp = bpy.data.objects["Lamp"]
```

We just assigned the three objects to three variables, camera, cube and lamp. To see the contents of a variable, type just its name:

```
cube  
bpy.data.objects['Cube']
```

```
camera  
bpy.data.objects['Camera']
```

```
lamp  
bpy.data.objects['Lamp']
```

Sometimes it's useful to use Python's *dir()* function to get more information about an object. For example

```
dir(cube)
```

will write the names of all functions and properties of the object. Quite a lot. But don't worry, soon you will know how to use all of them. You also may want to find out the type of something, which you can do like this:

```
type(cube)
```

In this case, just typing "cube" already displays the type, but from within an actual script, you would use *type()*. Something else which can be useful is viewing the documentation of Python objects. To do so, use the *help()* function on a variable or object.

```
help(bpy.data.objects)
```

This will print the documentation of the `bpy.data.objects` function we used. Of course, an easier way to view the documentation is the online HTML help. Click on **Help->Python API Reference**. Hopefully now your browser opens and displays the online documentation of the Blender Python API. If not, you should find it also here:

http://www.blender.org/documentation/blender_python_api_2_59_2/

In the documentation, click on `bpy`, then on `data` and you can see more examples. Using the documentation will get absolutely vital whenever you need to do something in a script not covered in a tutorial. And you will need to do so, else you wouldn't want to learn scripting at all.

Another resource you will need, depending on how far you will go with scripting, is the Python reference:

<http://docs.python.org/>

For this tutorial, maybe read the "Tutorial" section in the python docs, but you will understand everything without doing so.

Now, let's try to find out more about our cube. Type:

```
cube.type
```

It will tell us that the cube really is a Mesh object in Blender. Look up "type" in the online docs. Since the variable `cube` holds an Object, and "type" is an attribute of that Object, click on Object. There you find its "type".

Now that we know that the cube is a mesh, let's find out more about the mesh.

```
cubedata = bpy.data["Cube"]
```

'Note: in Blender 2.6.0 the script command actually looks like this, but the above is still valid: (Chronus001)

```
cubedata = bpy.data.meshes['Cube']
```

Every Blender object has data assigned to it, depending on the type. In the case of a mesh, the data are of type Mesh. In the documentation, go to the top again, and look for the Mesh module. It will contain documentation for the Mesh type. You can also try

```
dir(cubedata)
```

to get an idea about the available functions and properties. Try these:

```
list(cubedata.vertices)
list(cubedata.faces)
```

The first line will list the 8 vertices of the cube's mesh. The second line will list its 6 faces.

To get a member out of a list, you specify the index in square brackets, starting with 0. So:

```
v = cubedata.vertices[0]
```

This will assign the first vertex of the cube to the variable `v`. By now, you already know how to use `dir()` to get a list of possibly interesting things in `v`, find out about its type with `type()`, and where to look for the API documentation. It is in the module `Blender/Mesh`, when you click one "MVert" under "Classes".

```
v.co
```

This will display the 3D coordinates of the first vertex. Now, what if we want to know the coordinates of all vertices? We could of course assign them all to a variable, but the real way to do this is using a looping construct. There are numerous ways to do this, but one simple way looks like this:

```
for v in cubedata.vertices: print(v.co)
```

The **for variable in list:** construct assigns each element of the list to the variable in turn, and then executes the commands after the colon with the variable having the value of one particular list element. In a real script, you will have much more than a single command after the colon - so you would write them in the following lines.

By now, you should know enough to try yourself at a real script in the next section.

130.2. Creating a script

You can write scripts either in an external text editor, or in Blender's built in text editor. Move to a panel you wish to change to be the text editor, click the editor select button and choose "text editor" (shortcut) **SHIFT+F11**. Click the + New button at the bottom. If you want, you can enable line numbers and syntax coloring with the buttons at the bottom. Create a new script with *File* → *New*, paste the code below into it, and save it. Or alternatively, paste the code below into a file, and open that file with *File* → *Open* in Blender. As name choose something with the extension `.py`, for example `wikibooks.py`. Put it into Blender's user scripts path.

For different operating systems this is:

- Linux: `~/blender/scripts`
- Windows XP: `C:\Program Files\Blender Foundation\Blender\blender\scripts`
- Windows XP (alt): `C:\Documents and Settings\USERNAME\Application Data\Blender Foundation\Blender\blender\scripts`

- Windows Vista: C:\Users\USERNAME\AppData\Roaming\Blender Foundation\Blender\blender\scripts
- Mac OS X:
 - Under Mac OSX the path is actually hidden in the blender.app so to know the path you would have to know that the script folder is actually hidden in the blender.app itself. Assuming that Blender is in the applications directory the path would be "/Applications/blender/blender.app/Contents/MacOS/.blender/scripts" If you try to open the .app contents from the finder you will notice that .blender section of the path is not visible, while blender will still be able to navigate to this folder.
 - Right-click (or ctrl-click) the file "blender", and select "Show Package Contents" in the popup-menu. It will display all the hidden files under blender's folder, and select "scripts" folder inside it.
 - To see this folder from the OSX terminal use the ls -a command (lists all folders/files even hidden) in the MacOS folder of the listed path. It is probably a good idea to create an alias to the scripts folder in the "/Applications/blender-2.37a-OSX-10.3-powerpc" folder so that scripts can be easily manipulated through the finder. I know that its confusing that Blender should have its script folder buried inside the app but it is necessary to keep the app portable and not require an install.
 - A safer approach than the one above consists in keeping your scripts somewhere in your home folder: with this scheme, there is no risk of deleting your scripts when you upgrade your blender application, as they are not contained within its folder. A method that follows this principle is as follows: create a folder that will contain your scripts (or some of them) inside your own home directory; then, instead of putting your files directly in the .../.blender/scripts/ folder discussed above, simply add a *link* to your script directory in the .../.blender/scripts/ folder (for instance with the "ln -s" Unix command, or by doing "open /Applications/blender-2.37a-OSX-10.3-powerpc/blender.app/Contents/MacOS/.blender/scripts/" [adapted to your version of blender] and then creating a link through the Finder, with File->Make Alias). Blender will now find all the scripts that you put in your home directory: it will follow the link you created in its .../.blender/scripts/ folder and go to the corresponding folder in your own directory, and find all the python scripts you put there.

```

<nowiki>
#!BPY

"""
Name: 'Wikibooks'
Blender: 259
Group: 'Export'
Tooltip: 'Wikibooks sample exporter'
"""
import Blender
import bpy

def write(filename):
    out = open(filename, "w")
    sce= bpy.data.scenes.active
    for ob in sce.objects:
        out.write(ob.type + ": " + ob.name + "\n")
    out.close()

Blender.Window.FileSelector(write, "Export")
</nowiki>

```

Now, go back into the scripts window, and in its menu, click *Scripts* → *Update Menus*. If you saved it into the right path, from now on there should be an entry "Wikibooks" in the *File* → *Export* menu.

Try exporting any scene with it. It should open the file chooser dialog, and after you select a file and press the "Export" button, write a list of all objects in the scene into it. There will be one object per line, with the type, followed by a colon and the name.

How does it work? If you look at the script, you probably already know. But just in case, let's look at the script line by line. The first line contains this:

```
#!BPY
```

It tells Blender that this is a Blender script, and therefore it will consider it when scanning for scripts. Next simply follows a string, enclosed in triple quotation marks, so it can span multiple lines.

```
"""
Name: 'Wikibooks'
Blender: 259
Group: 'Export'
Tooltip: 'Wikibooks sample exporter'
"""
```

It contains four items, which Blender uses to place the script into its menus. The name, group (submenu name), and tooltip, all enclosed in single quotes. And the Blender version this is for. Note the group name must be one of those predefined by Blender (check the submenus in its Scripts menu to see the valid names); if Blender doesn't recognize the group name, the script will be put into the "Misc" submenu.

```
import Blender
import bpy
```

Remember how we said all functions from the bpy module start with "Blender."? In the interactive shell, we could simply use them, but in a python script, all used modules must be declared with an import statement (if you want to directly use functions from the Blender module in a script, you can simply replace the import statement above with "from Blender import *": no "Blender." prefix is necessary any more; however, this slows down the loading of your script). So the above simply allows us to use the functions from the Blender module in our script.

the bpy module is new and will replace Blender for data access.

```
def write(filename):
```

This defines a function in Python. The syntax is **def name(parameters):**. In our case, the name is "write", and we have one parameter, called "filename".

```
out = open(filename, "w")
```

Here we open a file for writing (the "w"), with the name passed to the function (filename). The python function "open" will open the file, and return a reference to it, which we store in the variable "out".

```
sce= bpy.data.scenes.active
for ob in sce.objects:
    out.write(ob.type + ": " + ob.name + "\n")
```

These three lines are our real export script. You already know what the first line does - first we get the current scene, then get a list of all objects in that scene, the for loop is assigning each one in turn to the variable "ob". The second line writes to the file - first the type of the object, then the string ":", then the name of the object, and finally a newline.

```
Blender.Window.FileSelector(write, "Export")
```

This is where execution of the script starts. It is simply a call of a Blender function (look it up in the API docs), which opens the file selector. It will display an "Export" button, and when the user clicks it, our function "write" from above gets called and is passed the selected filename.

This script isn't really very useful yet, but it shows the basics. You should now be able to e.g. also list all the materials in the scene. (Hint: They are just like objects, try to find them in the API docs.)

In the next section, we will learn how to export additional information about objects to our text file.

130.3. Exporting a Mesh

Our export script lists the type and name of every object, but that's not very useful yet. If we want to load the exported data in another application, we need more. Let's try to export a mesh object in the OBJ format.

The example below is a cube in the OBJ file format.

```
<nowiki>
v 1.000000 1.000000 -1.000000
v 1.000000 -1.000000 -1.000000
v -1.000000 -1.000000 -1.000000
v -1.000000 1.000000 -1.000000
v 1.000001 1.000000 1.000000
v 0.999999 -1.000000 1.000000
v -1.000000 -1.000000 1.000000
v -1.000000 1.000000 1.000000
f 1 2 3 4
f 5 8 7 6
f 1 5 6 2
f 2 6 7 3
f 3 7 8 4
f 5 1 4 8
</nowiki>
```

Here is a simple obj export script that exports a selected mesh object, used to export the OBJ file above.

```
<nowiki>
import Blender
import bpy

def write_obj(filepath):
    out = file(filepath, 'w')
    sce = bpy.data.scenes.active
```

```

ob = sce.objects.active
mesh = ob.getData(mesh=1)
for vert in mesh.verts:
out.write( 'v %f %f %f\n' % (vert.co.x, vert.co.y, vert.co.z)
)

for face in mesh.faces:
out.write('f')

for vert in face.v:
out.write( ' %i' % (vert.index + 1) )
out.write('\n')
out.close()
Blender.Window.FileSelector(write_obj, "Export")
</nowiki>

```

This script will export an OBJ file that can be read by many applications. Let's look at what's going on.

```

<nowiki>
sce = bpy.data.scenes.active
ob = sce.objects.active
</nowiki>

```

Here we are getting the object you last selected in the current scene. This will raise an error if there are no selected objects, but its an easy way to test a new exporter.

```

<nowiki>
mesh = ob.getData(mesh=1)
</nowiki>

```

This gets the objects linked datablock. At the moment we don't know its a mesh, another case where error checking would need to be added.

```

<nowiki>
for vert in mesh.verts:
out.write( 'v %f %f %f\n' % (vert.co.x, vert.co.y, vert.co.z)
)
</nowiki>

```

Here we write a line for every vertex, using string formatting to replace the "%f" on the left, with the 3 values on the right.

```

<nowiki>
for face in mesh.faces:
out.write('f')

for vert in face.v:
out.write( ' %i' % (vert.index + 1) )
out.write('\n')
</nowiki>

```

In the OBJ format each face references a number of vertex indices. For every face we have a line starting with "f", then loop through the vertices in the face. Just as mesh.verts are a list of all the vertices in a mesh, face.v is a list of verts in the face limited to 4 vertices maximum. (where mesh and face are arbitrary variable names assigned to Mesh and MFace objects) Every vertex writes its index on that same line with 1 added. This is because with the OBJ file format the first vertex is indexed at 1, whereas with Python and Blender the first item in a list is 0.

A new line is written so the next face will start on a new line. - in python '\n' represents a new line when written to a file.

131. Import scripts

Information

Applicable Blender version: 2.49

Importing objects into Blender is not that different from exporting. However, there are a few additional things to take care of. Firstly, all references to "export" in the header should be changed to "import". Secondly, instead of simply writing out data that Blender provides to us, we are responsible for giving data to Blender and ensuring that it is properly formatted. Although Blender is flexible, allowing us to ignore things like vertex indices, we do need to be careful that we do things in a sensible order.

Additionally, there is a bit of housekeeping to deal with. We should be in edit mode while modifying the mesh data. We also need to link up our newly created data to the scene, after it has been properly constructed, so that Blender can see it and maintain it. This makes it visible to the user, as well as ensuring that it gets saved along with the scene.

131.1. Importing a Mesh

Here is a simple script that can import an OBJ file created by the export script¹.

```
<nowiki>
import Blender
def import_obj(path):
    Blender.Window.WaitCursor(1)
    name = path.split('\\')[-1].split('/')[-1]
    mesh = Blender.NMesh.New( name ) # create a new mesh
    # parse the file
    file = open(path, 'r')
    for line in file:
        words = line.split()
        if len(words) == 0 or words[0].startswith('#'):
            pass
        elif words[0] == 'v':
            x, y, z = float(words[1]), float(words[2]),
float(words[3])
            mesh.verts.append(Blender.NMesh.Vert(x, y,
z))
        elif words[0] == 'f':
            faceVertList = []
            for faceIdx in words[1:]:
                faceVert = mesh.verts[int(faceIdx)-1]
                faceVertList.append(faceVert)
            newFace = Blender.NMesh.Face(faceVertList)
            mesh.addFace(newFace)

    # link the mesh to a new object
```

¹ Chapter 130 on page 1045

```
    ob = Blender.Object.New('Mesh', name) # Mesh must be spelled
just this--it is a specific type
    ob.link(mesh) # tell the object to use the mesh we just made
    scn = Blender.Scene.GetCurrent()
    for o in scn.getChildren():
        o.sel = 0

    scn.link(ob) # link the object to the current scene
    ob.sel= 1
    ob.Layers = scn.Layers
    Blender.Window.WaitCursor(0)
    Blender.Window.RedrawAll()

Blender.Window.FileSelector(import_obj, 'Import')
</nowiki>
```

This will load an OBJ file into Blender, creating a new mesh object. Let's take a look at the more interesting portions.

```
<nowiki>
Blender.Window.WaitCursor(1)
</nowiki>
```

Turn on the wait cursor so the user knows the computer is importing.

```
<nowiki>
name = path.split('\\')[ -1].split('/')[ -1]
mesh = Blender.NMesh.New( name ) # create a new mesh
</nowiki>
```

Here, we create a new mesh datablock. The name is made from the path only with the filename.

```
<nowiki>
ob = Blender.Object.New('Mesh', name)
ob.link(mesh)
</nowiki>
```

Next, we create a new object and link it to the mesh. This instantiates the mesh.

```
<nowiki>
scn = Blender.Scene.GetCurrent()
scn.link(ob) # link the object to the current scene
ob.sel= 1
ob.Layers = scn.Layers

</nowiki>
```

Finally, we attach the new object to the current scene, making it accessible to the user and ensuring that it will be saved along with the scene. We also select the new object so that the user can easily modify it after import. Copying the scenes layers ensures that the object will occupy the scenes current view layers.

```
<nowiki>
Blender.Window.WaitCursor(0)
Blender.Window.RedrawAll()
</nowiki>
```

Now the finishing touches. We turn off the wait cursor. We also redraw the 3D window to ensure that the new object is initially visible. If we didn't do this, the object might not appear until the user changes the viewpoint or forces a redraw in some other way.

132. Scripts for modifying meshes

(To be written.)

See [saltshaker¹](#) a basic but functional Python script for Blender. The page includes details of how it was made.

http://jmsoler.free.fr/didacticiel/blender/tutor/python_script01_en.htm² is a good website for learning about mesh creation.

¹ <http://saltshaker.sourceforge.net/>

² http://jmsoler.free.fr/didacticiel/blender/tutor/python_script01_en.htm

133. Advanced Modeling Introduction

133.1. Advanced Modeling

We have learned quite a few useful techniques that we can now use to make something really cool or fun. But there are plenty of little tricks hidden within Blender that will allow us to make very realistic or intricate 3D graphics with a much lower amount of memory used. It usually takes an extra application or two, and the more advanced the program, the better; however, in the next few tutorials the *Paint* program--or its equivalent on another OS--will be just fine for our purposes.

The purpose of these modeling tutorials will be to lower the poly count on your models--this could range from no effect to cutting out 99.99% of your original faces, and still being able to keep the realistic look you were going for. Make sure you have read over the other tutorials in this book or at least are very familiar with Blender, so you can better understand how to work these modeling techniques.

134. Advanced Modeling

134.1. Making Better Models

This section will deal with the actual modeling of objects and how to make better, more detailed objects much faster and more efficient. It will also deal with making more accurate models, whether it is cartoonish or realistic. Either way, this tutorial will deal with making better models and using the more advanced modeling systems in Blender.

135. HDRi

135.1. Introduction

You may have heard various people talk about HDR images. (WETA, Lucas, even Tim Sweeny). HDR images are part of a technology called HDRi which stands for "High Dynamic Range (image)". So... what on earth does that mean?

Here's a link to Wikipedia's article on the HDR format¹ which I personally give all my credit to Paul Debevec for putting it to use for computer graphics purposes. Anyway, before you start trying to understand the usefulness of HDRi, please read the wikipedia link.

Also, visit Paul Debevec's website² if you've got some more time to spare.

To sum up the excitement of HDR CG, think of it like the hype of the next-generation videogames that are about to come out, except set the stage for 1996 instead of 2006. Paul Debevec pioneered parallax mapping, HDR lighting, image-based modeling, his latest work includes some even more amazing technologies, and for the record he's my hero too.

To use HDRi images for 3D rendering, you need something called a light probe³...

135.2. Definitions

HDRI

1 http://en.wikipedia.org/wiki/High_dynamic_range_imaging

2 <http://www.debevec.org>

3 Chapter 136 on page 1071

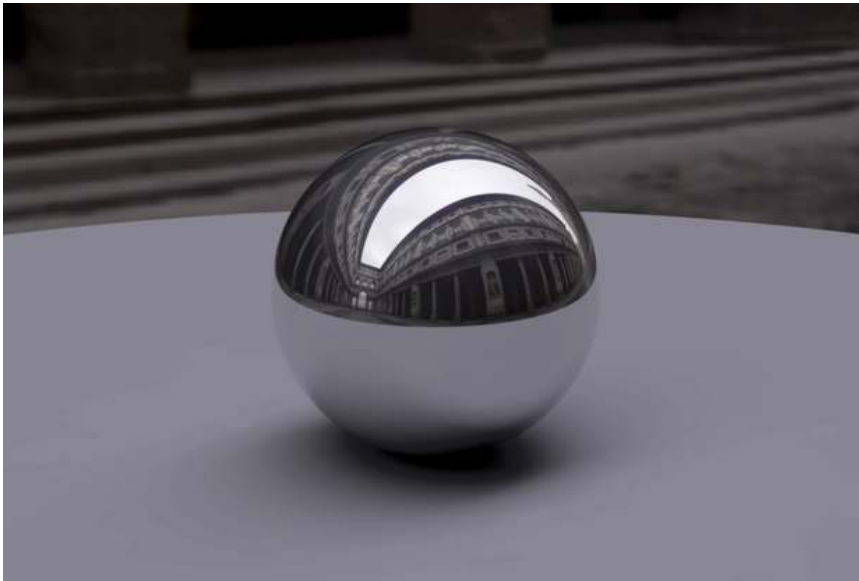


Figure 916 HDR image file example

HDRi stands for **High Dynamic Range Imaging**, and is basically an image format that contains from the deepest shadow up to the brightest highlight information. While an 'ordinary' digital image contains only 8 bits of information per color (red, green, blue) which gives you 256 gradations per color, the HDR image format stores the 3 colors with floating point accuracy. Thus the 'depth' from dark to light per color is virtually unlimited. Using HDR images in a 3D environment will result in very realistic and convincing shadows, highlights and reflections. This is very important for realistic emulation of chrome for example.

Light Probe

A Light Probe is a HDR image containing 360 by 360 degrees (solid angle⁴ 4π steradians⁵) image information. In other words : it's a 360 degree spherical panorama image, not only looking around the horizon, but also up and down. Thus a Light Probe image contains all visible information as can be seen from a specific point, wherever you turn your head.

135.3. Usage

Given that a Light Probe image is an 'all around' image with a high dynamic range, it's the perfect solution for your 'world' background, especially for a 3D animation.

4 http://en.wikipedia.org/wiki/Solid_angle

5 <http://en.wikipedia.org/wiki/Steradian>

135.4. Quick Tutorial (*for experienced blendies*)

First of all, you'll need an HDR image. There is a whole range at <http://debevec.org/Probes/>⁶ that you can download for free. (There are even more at <http://blenderartists.org/forum/showthread.php?t=24038>⁷). I will use the St. Peter's Basilica probe, but any other HDR image will do just fine.

To apply the HDRI environment to your scene:

- Go to the shading settings (press F5) and click the World button.
- Enable "Real" to force the horizon to stay still, as opposed to follow the camera
- If you're rendering with Blender Internal (ver. 2.49), go to "Ambient Occlusion" tab and activate "Ambient Occlusion", then activate "Sky Texture". Skip this step if you're using external renderers.
- In the "Texture and Input" tab, click "Add New" and "Angmap".
- Then go to the "Map To" tab and deactivate "Blend" and activate "Hori".
- Now go to the Texture settings (press F6) and change the "Texture Type" to "Image".
- Click the "Load Image" button and locate your HDR image.
- If you're using YafRay or other external renderers, you need to turn on Global Illumination and to set the quality to something other than "none".

135.5. Step-by-step Tutorial

135.5.1. Build a simple scene

To see the advantage of using a 360 by 360 world image, the simplest example to demonstrate this is a scene with a mirrored sphere.

⁶ <http://debevec.org/Probes/>

⁷ <http://blenderartists.org/forum/showthread.php?t=24038>

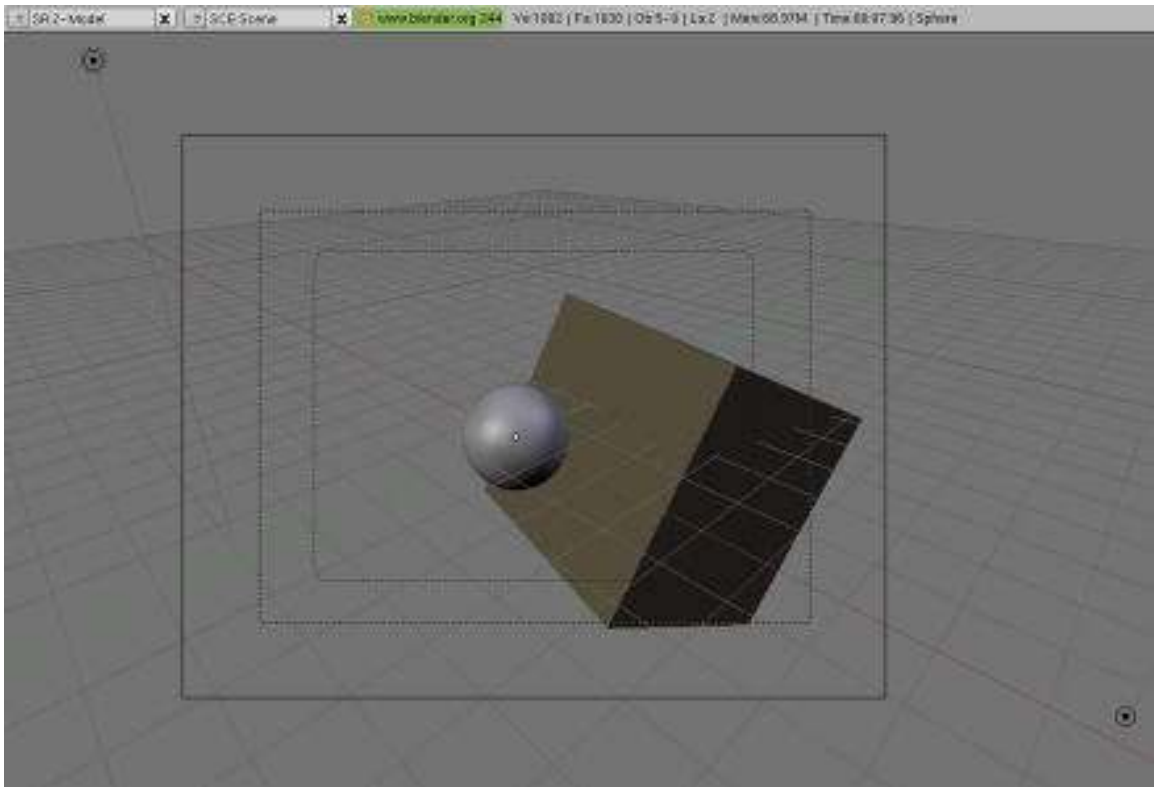


Figure 917 Scene

1. Add a sphere and a cube to your scene and place them in a bit of an interesting position. (note that I added a second lamp to light up the shadow part of the cube)
2. Perhaps give the cube a different color than the default grey.

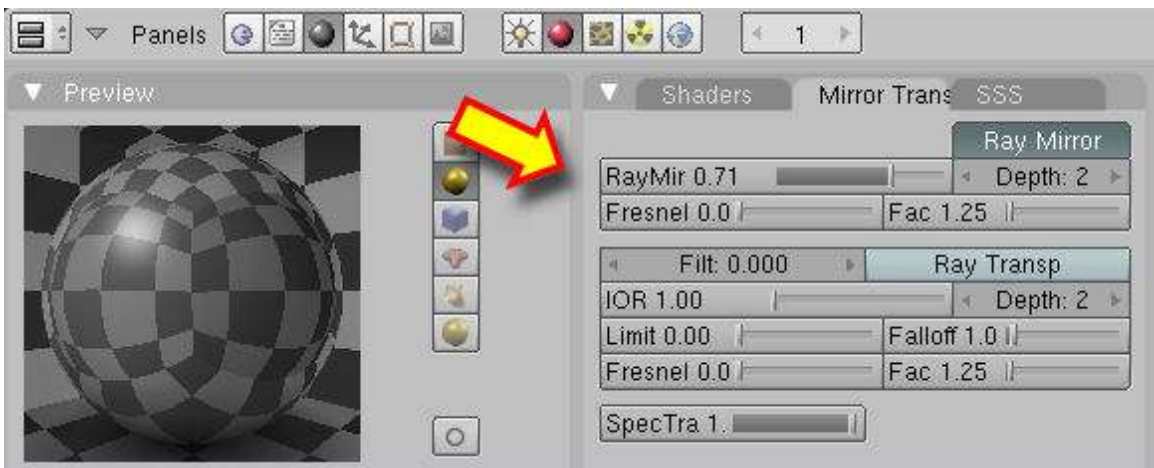


Figure 918 Material setting for a mirror surface and a preview of it

1. Give the sphere a mirrored material : go to the **Shading -> Material** panel (F5) and find the **Mirror Trans** buttons.
2. Check that **Ray Mirror** button is pressed. If it is not, check it.

3. Set the **RayMir** value to a value of 0.5 or higher. Your preview should show the reflection of the checkboard environment.

135.5.2. Render with HDR (Blender Internal v2.49)

1. Download a HDR image (see Paul Debevec's website)
2. Go to the shading settings (press F5) and click the World button.

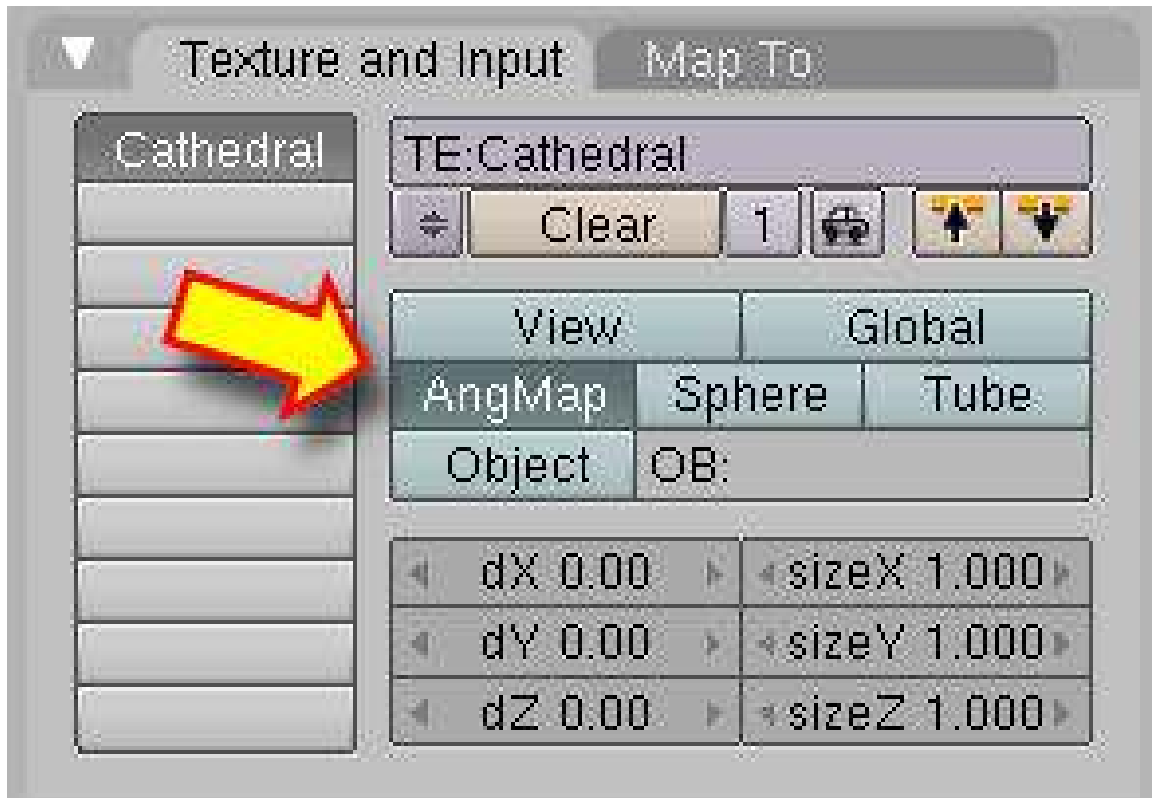


Figure 919 AngMap enabled

1. Go to "Ambient Occlusion" tab and activate "Ambient Occlusion", then make sure "Raytrace" is chosen as gather method, and activate "Sky Texture"
2. Adjust quality settings:
 - a) Increasing number of samples reduces noise, but increases render times
 - b) Adaptive QMC is faster, but generates more noise than Constant QMC
3. In the "Texture and Input" tab, click "Add New" and "Angmap"
4. Then go to the "Map To" tab and deactivate "Blend" and activate "Hori".
5. Now go to the Texture settings (press F6) and change the "Texture Type" to "Image".
6. Click the "Load Image" button and locate your HDR image.

135.5.3. Render with YafRay

1. Download a HDR image (see Paul Debevec's website).

2. Go to the shading settings (press F5) and click the World button.
3. In the "Texture and Input" tab, click "Add New" and "Angmap". Note that the Angmap button is the important thing to tell Blender that this file is a Light Probe file !
4. Then go to the "Map To" tab and deactivate "Blend" and activate "Hori".
5. Now go to the Texture settings (press F6) and change the "Texture Type" to "Image".
6. Click the "Load Image" button and locate your HDR image.

(optional step, as it was not needed for my setup :)

5. Press F10 and change the "Blender Internal" to "YafRay". You need to turn on Global Illumination and to set the quality to something other than "none". Note that the YafRay renderer does not come standard with the Blender installation. You need to download and install this separately.

Your result should look like this: (Rendering: left with Blender, right with YafRay) [Click for larger version](#)

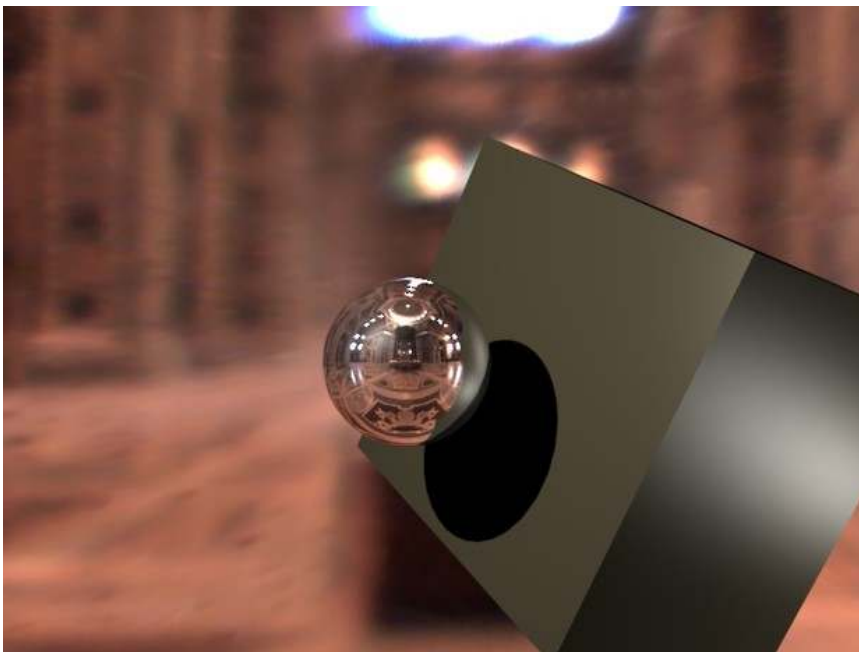


Figure 920

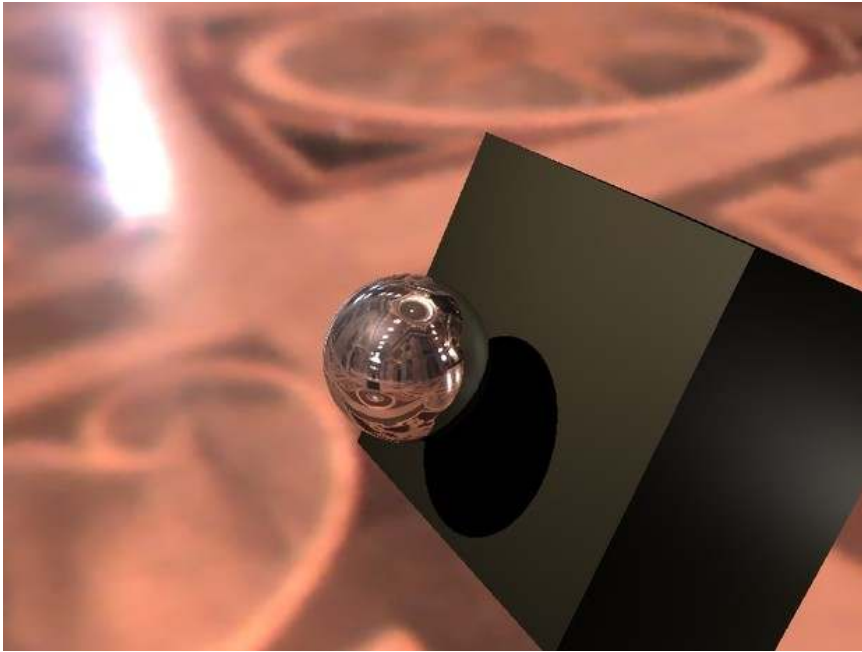


Figure 921

Note that the reflecting ball reflects the whole interior from every angle, even though we added just a single image to the World settings !

136. Creating a Light Probe

The light probe is, in the simplest terms, a photograph of your environment. they work in very much the same way that reflection maps do, and are made the same way.

Equipment:

1. A pure silver ball. Try a plastic Christmas tree ball ornament.
2. A camera, prefferably digital. If you have a high-end digital camera, you'll have less work ahead.
3. A place you'd like to capture the lighting from. Try laying different things in your environment to get a good idea.
4. Something to fire the shutter without touching the camera. For digital cameras, Paul Debevec recommends using a program that will take all the pictures for you.
5. A tripod.

Set it up like so. Remember that the height of your camera and the height of the ball relative to each other controls the angle at which the horizon will be shot. In other words, shoot the ball at the same angle that you plan to shoot your 3D scene in. If your scene is animated... consider making a similar rig except attaching your reflective ball to a video camera as was done for Flight of the Navigator¹.

¹ http://en.wikipedia.org/wiki/Flight_of_the_Navigator

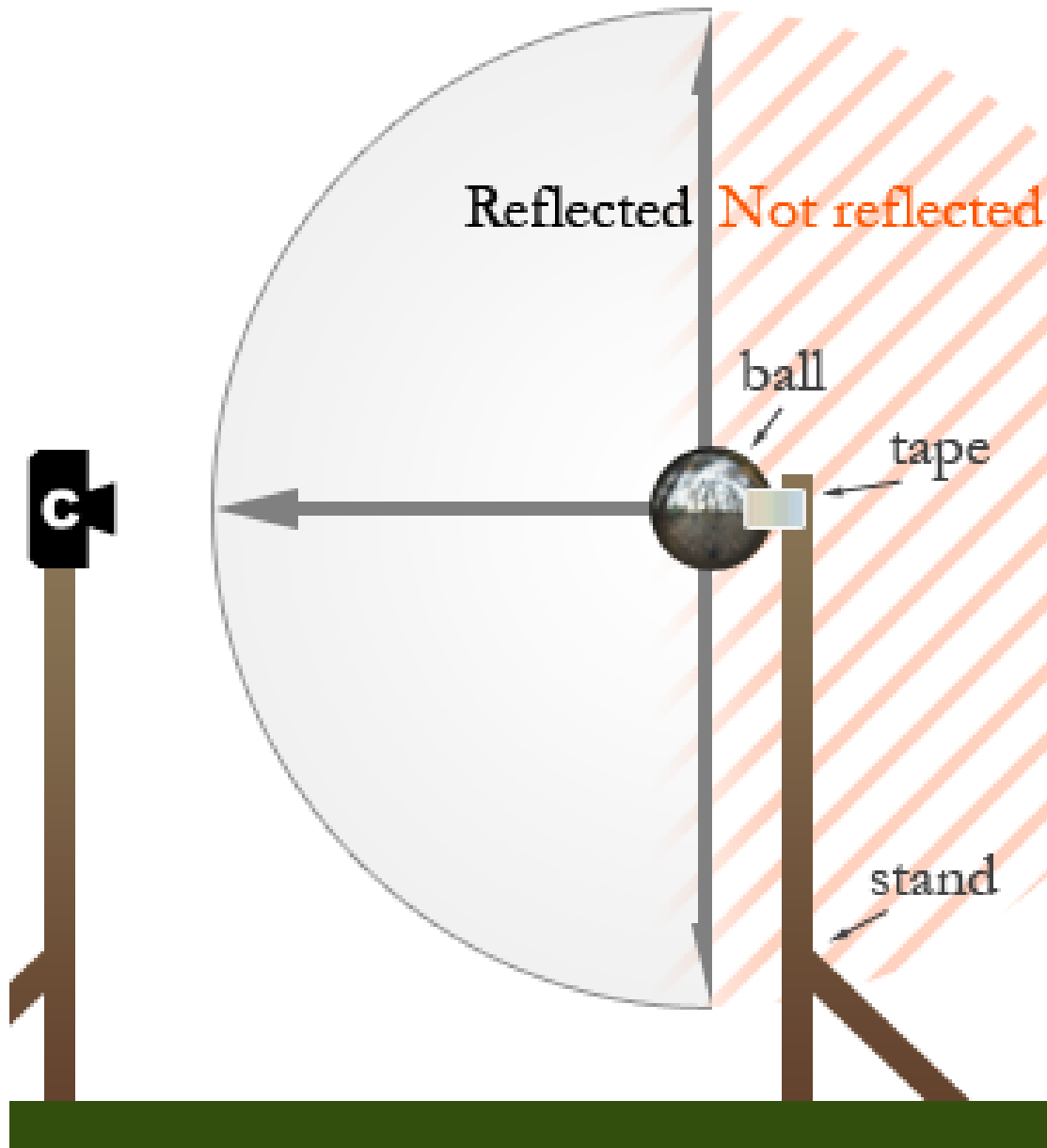


Figure 922

1. Note that the diagram is incorrect. The camera will see items reflected from behind the sphere as well.

The process:

1. Set up your rig and take a series of images with varying exposure times.
2. Taking 2 sets of pictures, each offset by 90 degrees, will enable you to get better coverage of the background and eliminate the reflection of the camera taking the picture.

Good article²

² <http://gl.ict.usc.edu/HDRShop/tutorial/tutorial5.html>

137. Making Landscapes with heightmaps

This tutorial will show you how to make advanced terrain such as mountains using Blender and gimp or any other image editing software. Blender has the ability to use height maps to create meshes. Height maps are black and white images with white representing the highest point and black the lowest.

137.1. Creating the heightmap image

Note: This entire Tutorial presumes that you are already familiar with other Editing Software, such as GIMP... and already know how to create "textures" with that software. However, you do not have to use another program to obtain a texture. At the end of the next section, they show you how to generate a random 'cloud' texture which you can use directly. Do not worry about GIMP or Photoshop. Just skip the rest of this section and most of the next one.

To begin with, open your image editing software. This part applies to Gimp only (downloadable at <http://gimp.org>),¹ if you use another program you will have to do it another way.

First use the "New" menu option to create an image 1600 wide by 1200 tall. Go to **Filters** → *Render* → *Clouds* → *Plasma* (In Photoshop, this is Filter > Render > Clouds). For this example just use the default settings, it doesn't really matter. Click OK. You should now have a nice colorful image. We don't want that, we want it in black and white. Make sure your Gradient is set to "FG to BG(RGB)"(this is the default anyway) and that your foreground color is black and the background color is white. Go to **Colors** → *Map* → *Gradient Map*. This will convert it to black and white for you.

Save your image as PNG or JPEG.

It should look something like this

¹ <http://gimp.org>,

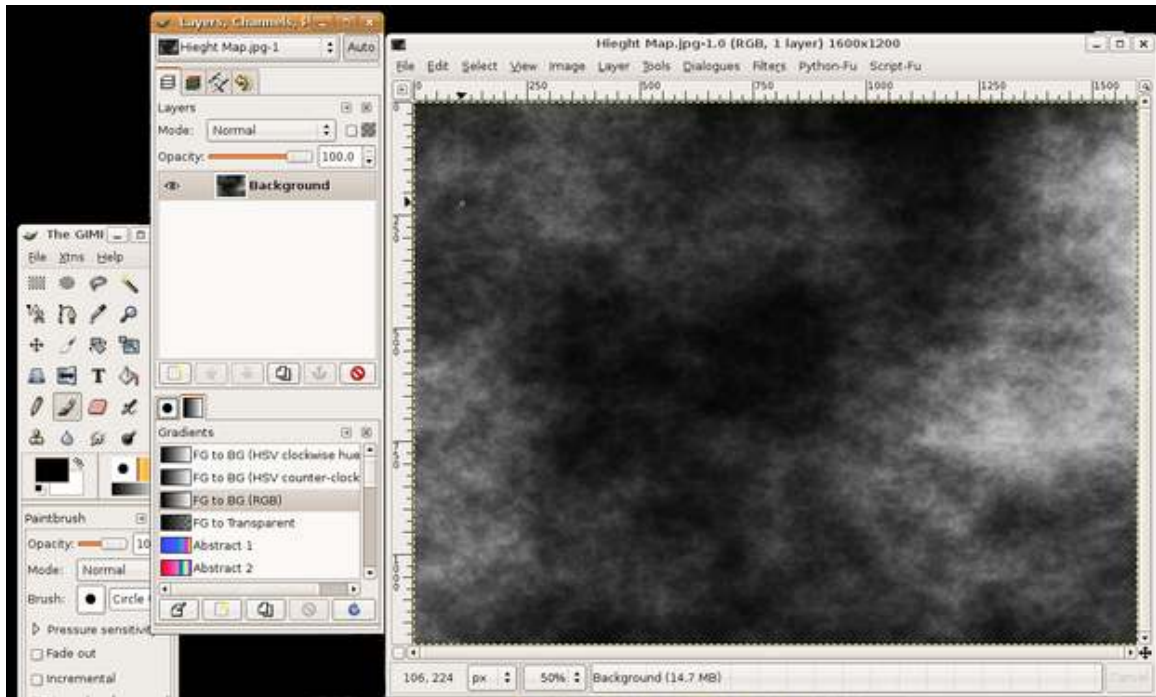


Figure 923

137.2. Create grid and add the image as texture

Open Blender and delete the default cube.

Add a grid (**SPACE** → *Add* → *Mesh* → *Grid*) with resolutions 32 and 32 from top view. Do not scale the grid just yet, zoom instead if you wish to take a closer look at your grid. It will be explained later why you shouldn't scale now.

Note: When I followed this tutorial it didn't work - the vertices in the grid were moving sideways rather than vertically. It turns out I needed to **TAB** into object mode after creating the grid, then hit **CTRL+A** to reset the transformation, then **TAB** back into edit mode. If someone could explain why that was necessary, or what it does exactly, that would be helpful. *Response:* You should create the basic shape and outline in object mode, then edit the details in edit mode, kinda like the names imply. *Response 2:* When you create an object using the Front or Side views while having **Aligned to View** selected in the **Edit Methods** in **Blender Settings Panel**, the Local Z axis of the object will be pointing to the Global Y or X axis, respectively. For the matter of this tutorial, if you do that, you'll be forced to rotate the object 90° in X or Y, before going on. That's where problems may begin, if you rotate it in **Edit Mode** which doesn't change local axis orientations. In that case you'll be forced to reset transformations (**Ctrl+A** → **Scale and Rotation to ObData**). However, if you rotate it in **Object Mode**, and if you make sure the Local Z axis points upwards after the rotation, you should be seeing good results later on. (*To see the Local orientations of an object, Press F7 for the **Object Panel** (the one with three arrows, 4th from the left) and in the **Draw** pane click the **Axis** button.*)

Click **F5** to go to the *Shading panel*. By default the *Material buttons* button should be selected (button with red ball), but if it isn't then click on it. If this window has a panel named *Preview* and it is empty then look to the right for the *Links and Pipeline* panel. Here you should find an *Add New* button beneath *Link to Object*, click on it. 3 more panels should appear when you do this. You should find the *Texture buttons* button, it is just to the right of the *Material button* and is black & yellow. You may also just hit **F6**. Now you should get two new panels titled *Preview* and *Texture*, in the *Texture* panel you'll find an *Add new* button, click it. Where the button just was it should now be a text field saying something like "TE:Tex.001", click in the text field and write "height" instead. Look a bit to the right and down for a dropdown menu allowing you to choose *Texture Type*, click it and select *Image*. Two more panels will appear and the rightmost should be *Image* with a *Load* button.

Note: The *Texture buttons* button is not part of the windows that pop up after you click *Add New* button, but it directly to the right of the *Material* button that is a red ball.

Note: If you can't find the *Texture Type* menu, press **F6**. Once you press **F6**, or the *Texture buttons* button, you'll see another panel, also called *Texture*. This one, however, shows a *Texture Type* menu after you select *Add new*. Having two very similar panels can be a nasty trap.

Before you load the image take a look at the *Map Image* panel. About in the middle there should be 5 buttons, *Extend*, *Clip*, *ClipCube*, *Repeat* and *Checker*. Select *Clip* or your heightmap may wrap around the grid (if this doesn't make sense then do this tutorial twice, one where you keep *Repeat* selected and one without. In that case make sure there's some bright white spots in your height map on each edge).

Click the *Load* button and load the heightmap you just created in your favourite image editing software.

If you're going for the random cloud-type image, though, you can make it from within Blender itself - when you add a texture, in the menu where you select 'image,' before actually selecting it, pick 'clouds' instead, and play with the settings.



Figure 924

Click the *Material buttons* button again (the red ball button) and look in the *Preview* panel. If this just shows a black ball then click on the uppermost button just to the right of the black ball.



Figure 925

137.3. Use texture as heightmap

(Blender 2.5+ note: Now you do this using a "Displace" modifier that applies the Texture you created.)

Look to the top right of the rightmost panel, the window should have 3 panes titled *Texture*, *Map Input* and *Map To*. Click on the *Map To* pane. A whole lot of buttons will appear:

- Turn off the *Col* button. "Col" stands for "color," and we don't want our plane to be *colored* by this texture.
- All of these buttons should be turned off: we're not going to be using this texture to "Map To" anything. Instead, we'll be using it to deform the actual geometry.

Go to the *Edit window* (**F9**) and enter edit mode (**TAB**) if you're in *Object Mode*. Go to either front or side view so you can see what you are doing. Click the *Noise* button in the *Mesh Tools*-panel (it's located in the top left corner of this panel) and your grid should start to change shape.

Note: make sure all your vertices are selected with **AKEY** or whatever you want to use to select. If not selected, *Noise* won't do anything!

- (*While you could, theoretically, use this to your advantage ... intentionally de-selecting vertices that you don't want to influence ... in practice that's quite awkward if for some reason you have to do it again. It's much easier to leave any such areas, that you don't want to influence, "completely white" in the image.*)

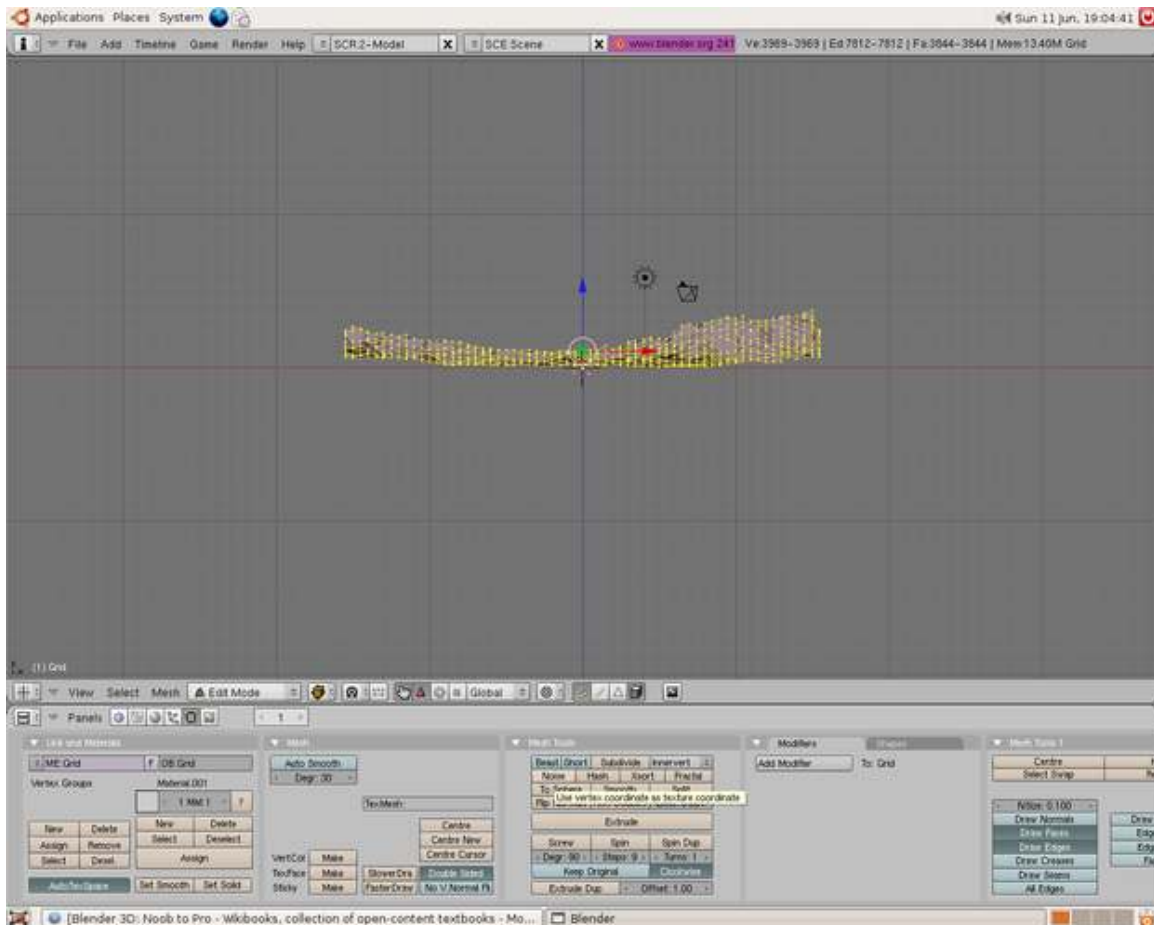


Figure 926

Continue clicking the button until the terrain has reached the height you want and voilà, you have just made terrain using heightmaps. You may want to subsurf the shape to get a smoother effect. If you wish to scale your new landscape then now is the time. There seems to be a bug or gotcha causing your heightmap to be tiled on your grid rather than resized if you scale the grid before applying the heightmap. Hopefully someone familiar with blender will clarify how this should be done properly, but until then this method will work.

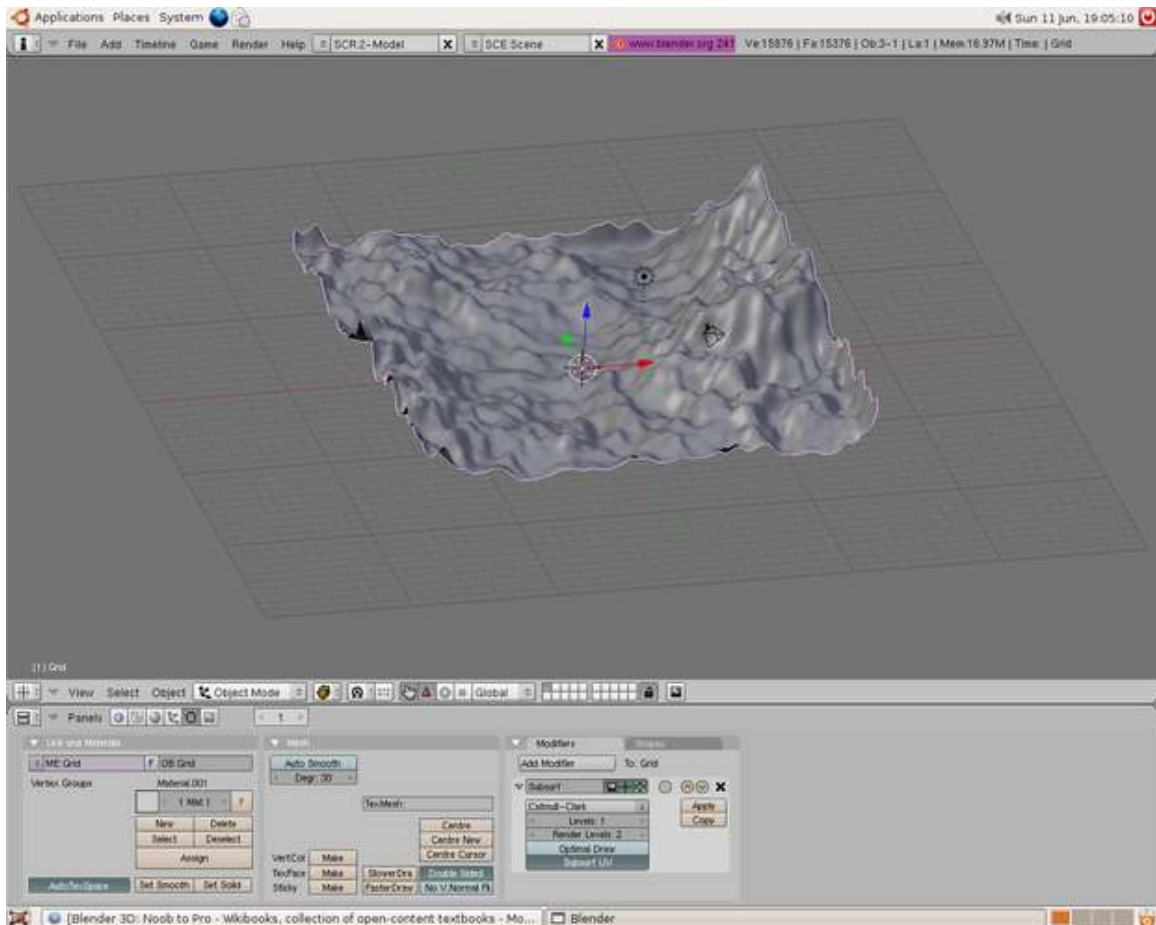


Figure 927

Tips:

- 1.If you want a more jagged landscape then adjust the contrast on the image editing software
- 2.Once you've finished using the *Noise* button in the *Mesh Tools* panel as shown above, the texture is no longer needed: the effect it has made on the geometry of the object is permanent. If texture-resources are tight, you can remove it now. But it's advisable to keep it, unchanged, in case you need to re-do your work at some time in the future. If you want to use other kinds of texture-maps (*Color*, *Normal* maps, and so-forth), you can use the landscape-image as a handy reference (say in a background-layer in your painting program) for correct placement of these features, but you should not alter the landscape-image itself).
- 3.Height maps can also be used to make city terrain. Draw white squares on a black background. If you do this make sure you make the squares perfectly parallel to the sides of the image. This prevents jagged edges to buildings. You can also use gray squares to adjust the buildings to different heights.
- 4.The "Noise" mesh-editing button displaces vertices in the object's Z-Axis and negative Z-Axis only. To deform your mesh's other dimensions, simply rotate your object and "apply rotation," or rotate the vertices in edit mode, and apply Noise. Then, rotate it back again to get your original orientation.

5.The "Noise" button permanently modifies your mesh according to the material texture. Each click adds onto the current mesh. For a temporary effect, map the texture to *Disp*(lacement) for a render-time effect (*and do not click the "Noise" button at all*). In object/edit mode your object will appear normal, but will render deformed.

6.*Note:* The "Noise" mesh-editing *button*, as used here, is not the same as the "Noise" *texture* that is referred to in other tutorials!

138. How to Do Procedural Landscape Modeling

138.1. Mountains

I was playing around in Blender several months ago and messing with cloud textures when I found a great way to make landscapes entirely within Blender (without height maps from The Gimp or some other application) and decided to make a tutorial (yes, it does use the noise tool).

138.1.1. Making the Mesh

To start, open up Blender and with the cube selected, press the **XKEY** and click Erase Selected Objects.

Add a grid (**SPACE**--> Add--> Mesh--> Grid), using whatever numbers you like, remembering that bigger numbers means more vertices. For this tutorial, I'll use 60 for x and y resolution. Scale the grid up, but only a little bit (I went up to 1.3) as the texture won't make very tall landscape with a very big grid. Press the **F5** key to go to the Materials tab, then add a new material.

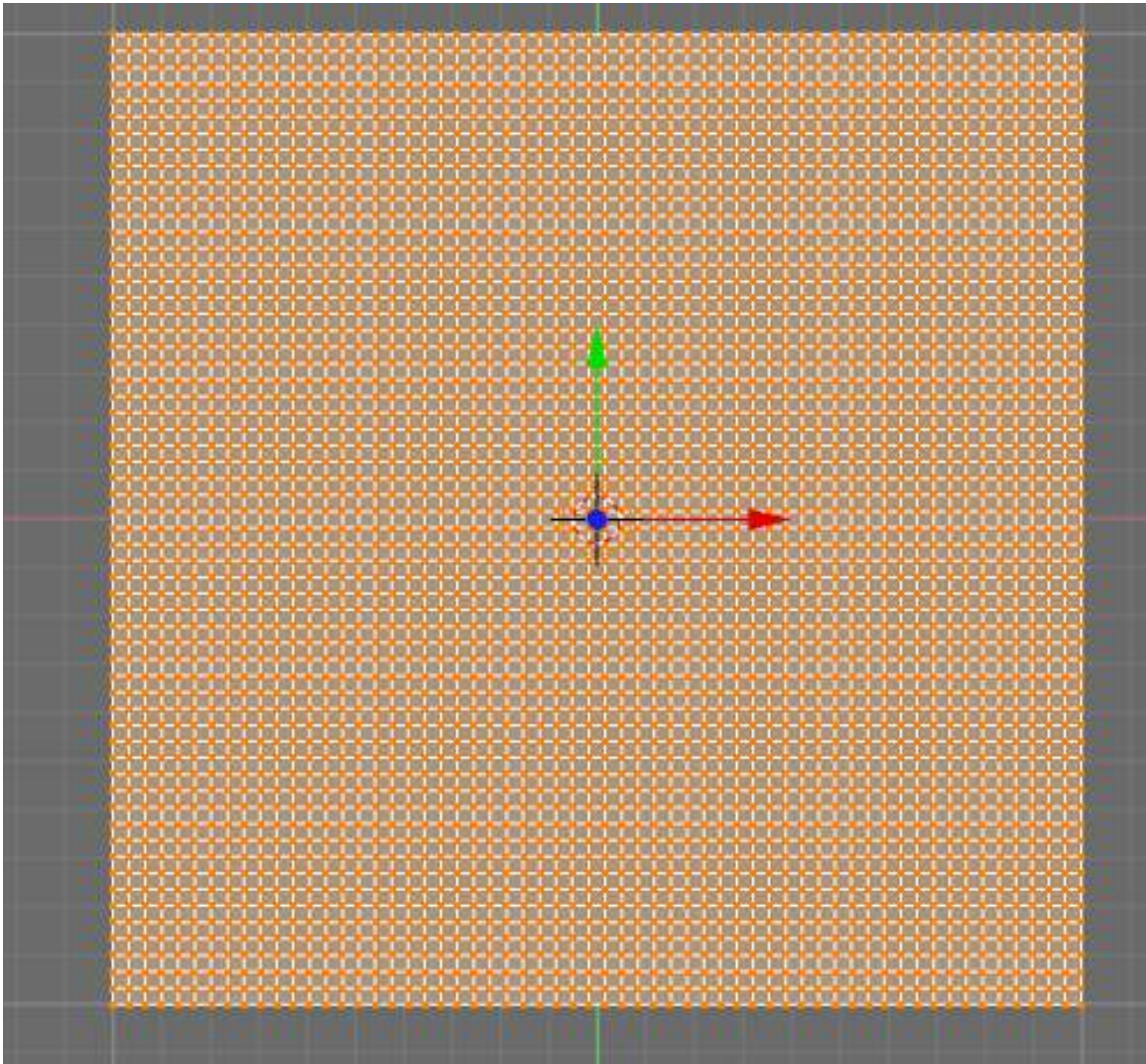


Figure 928

138.1.2. Displace and Shade

Now press the **F6** key to go the Textures panel, and add a cloud texture in the very first slot (slot 0).

After Pressing F6, look at the windows on the bottom. To the left is the "preview" window and next to that (to the right) is a *Texture* window with an *Add* button... click that and then go to the little pull down menu that shows up just below and to the right. From there, find and select "Clouds"... and now, continue!

You may be wondering how we are going to make landscape with a cloud texture, when you could just use the Gimp height map plugin, Terragen, or the A.N.T. modeling script right in Blender. Well, we are going to make the landscape with a **height map**. The color determines the height: white = very tall; grey = halfway tall; black = no height. This makes the cloud texture a very good candidate for land.

The next thing we are going to do is adjust the size. You can set the noise size to whatever you want, but I want to have several bigger pieces of landscape, instead of a bunch of little ones so I'm setting noise size to 1.00 and noise depth to 6, as I want a high level of detail. (The *Preview* immediately shows you the effect of your "tweaks" as you make them.)

Noob Note: You can hold shift and click the number next to noise size then just type 1 and press enter instead of using the arrows. The same applies for NoiseDepth, almost every other setting in blender changes like this as well.

At the top, we can select *Hard noise* or *Soft noise*. Each gives a different landscape, so experiment with them. Next, go to the *Colors* tab (it's right next to the texture button by default). Adjust the brightness and contrast as required. For "Hard noise," try a brightness of 1.9 and a contrast of 3.45. For "Soft noise," a brightness around 1.0 to 1.2 should be fine.

Noob Note: The above noob note applies to the brightness setting as well!

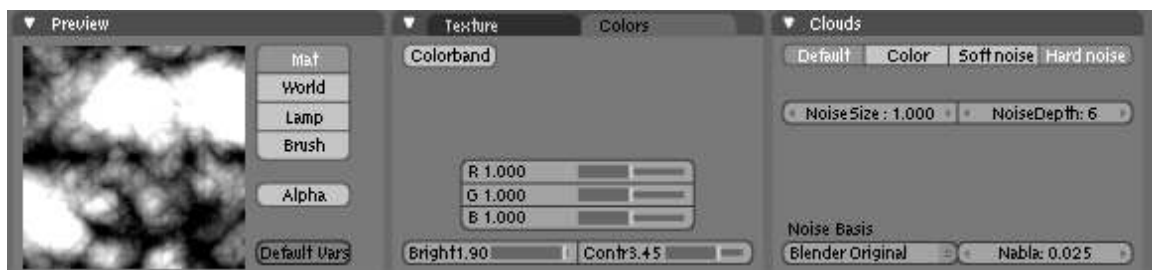


Figure 929

Jump back to material buttons with **F5** and go into the "Map To" panel. Turn off *col* and press *nor* once.

Noob Note: I had a difficult time finding the "Map To" so if you are too... This is one of the several small Windows that opens at the bottom of your screen after pressing **F5** and for me, it was on the far, far right, off my screen. I had to use my mouse wheel to scroll the Windows to the side to bring it into view and found it as a Tab on a Window with two other tabs. "Col" and "Nor" are buttons in the *Map To* Tab.

- *What the buttons are for:* The "Map To" settings determine which various attributes of the material will be affected by our "cloudy texture."
 - "Col" refers to "color." We turn it off here, because we don't want our surface to *look* "cloudy."
 - "Nor" refers to "surface normals." This is the angle at which light seems to reflect from the surface. (The so-called "bump map.")
 - Experiment with the other "Map To" buttons, as well! For instance, "Disp" (for "displacement") actually causes a map to deform the geometry of a surface... but doing so only during rendering.

In this tutorial, as you will see, we are going to *combine* several techniques to produce a very rocky surface. Here, we are changing the way that the surface reflects light. Then, we'll actually deform the surface geometry.

Turn the *nor* value all the way up, to 25 (that will give it a nice rocky look). Go into the *Shaders* tab and then change the "Diffuse Shader" (the top left selector, defaulting to "Lambert") to "Oren-Nayer," and the "Specular Shader" (the 2nd selector, defaulting to "CookTorr") to "Blinn." Adjust the *Rough*

value in Oren-Nayer to 1.5 (rock is really rough). Also, set the *Spec* value to 0.01 and *Hard* to 25. Now we are ready to make some mountains!

- *Trivia note:* In computer graphics, a "shader" is a mathematical function, and they're customarily named after the clever people who invented them. These functions determine exactly how the cloudy-texture will "map to" the attributes (e.g. "Nor"...) that we selected.



Figure 930

138.1.3. Making Mountains

Press **TAB** to go into edit mode and press the **AKEY** twice, in case you had any specific vertices selected. Go into the front view and press the "Noise" button (located under mesh tools in **F9**). You should see the vertices jump up a good bit.

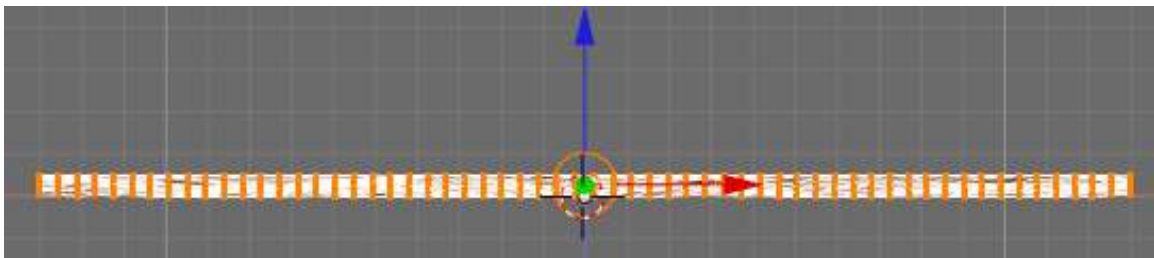


Figure 931

If not, you may have scaled your grid up too much, so make it somewhat smaller. Depending on how high you want your mountains, press the *Noise* button to your required amount, though I pressed it 8 more times. The mesh doesn't look very good right now, as it is not only blocky, but it is missing that last bit of random detail.

- **Important note!** The *mesh-editing button* named "Noise" causes a permanent change to the geometry of the object in the "Z"-axis, as provided by the texture. (The vertices actually *moved*, and their new position is permanent.)
 - There is also a *procedural texture* named "Noise," but that is just pure coincidence. The two are unrelated, both in what they do and in how they work.
- **Important note!** Those using Blender 2.5 Beta in order to get the height map to affect the geometry you will need to use the displace **modifier** instead.

If you are not in edit mode, go into it and select all the vertices. Press *Fractal* (in the same row as *Noise* under mesh tools) and enter a value of 15 in the random factor. Hopefully, your system can manage all these vertices: if not, then don't do any fractal subdivide. If you want to, you can use a smaller value in the fractal box and do it several more times for more displacement. Be careful, though: too much randomness will cause vertices to separate, causing tears in the mesh that you will have to fix by hand.

Finally, tab out of edit mode and press *Set Smooth*. Then, if you want to make the mesh even more smoother, add a *Subsurf* modifier, with what ever level you want. (I'm going with 1 for faster render times.)

If you desire, you can scale up the mesh however big you want though for close-ups, you may want another level of subsurf or another subdivide. Then, apply colors, set up the lights, and render! A word to the wise, however: this process creates a huge file (about 986kb).

138.1.4. Discussion

You have just made mountains all in Blender, without having to generate height maps in The Gimp, or any other program. The next step is to go beyond the tutorial. Try using different noise sizes and noise basis, and even try using other textures like musgrave and marble. Or try using two textures and see the conclusion. The possibilities are almost infinite! Have fun.

Observe how the technique illustrated *this* tutorial is "the same, yet different" from the one demonstrated in the tutorial for "height maps":

- A *procedural texture* was used to supply changes to the *Normal Mapping* of the object at render time. (This is the angle at which the object reflects light.) (The "height mapping" tutorial did not do this.)
- Then, the *mesh-editing button* (also...) named "Noise" was used to deform the actual geometry of the mesh. (This *is* the same technique used in the "Height Map" tutorial, although a different kind of texture .. an image .. was used.)
- The texture, once used for the height-mapping (deformations) was left in place, still mapped to the surface "Nor"mals. The result will be a *very* mountainous landscape.

Yep! The techniques shown in all of these terrain-modeling tutorials *can* be combined! And, they frequently are. Very often the most satisfactory results are obtained by combining several different techniques. You could have used an image-map (from the "Height Mapping" tutorial) to deform the

geometry, then used the "noisy *Normal*-mapping" idea from the first part of this tutorial, and even used smoothing. All at the same time. The choice is yours.

When using "Map To," remember that a single material can have several different textures applied to it, each one mapped to the same or to different attributes. This mapping can even be *animated*.

Experiment! The sky's the limit!

139. Landscape Modeling I: Basic Terrain

I worked with a group to make a small animation for class and I was responsible for the environment modeling. I really liked the results of the process I used, so I decided to share it with others. By the end of this tutorial you'll have the know-how to create your own flexible, realistic terrain utilizing multiple textures for different ground types. This tutorial assumes that you have the very basic understanding of using Blender (how to add/remove a mesh, how to change views, etc...) I use the following textures in this demo. Feel free to use them if you don't have something on hand to use.



Figure 932

NOTE: This is not the actual texture I used in this demo. I couldn't find the original source for the one I used (though I know it was a public image), so to avoid licensing issues I replaced it with one that I did have licensing info on. Sorry.

139.1. Creating the Canvas

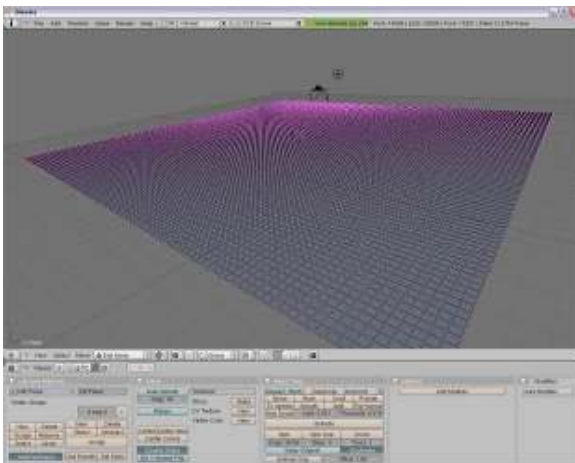


Figure 933 The grid that will be used to build our landscape.

The "canvas" that we'll use for our terrain is an evenly spaced grid of vertices. Open a new project, delete the default cube, and add a grid mesh. Use whatever size you want. The more vertices you have, the more detailed and realistic your landscape will appear, but don't get crazy with it since we'll apply subsurfacing at the end to smooth it out. At the same time, you do want enough vertices to prevent sharp edges, and we'll need them in the second tutorial when we cover texture stenciling. So consider how big you want your landscape and try to find a reasonable balance. I like to start with 100x100. The grid will be pretty small, so scale the entire thing up, let's say by a factor of 20. This grid will be used to build our landscape by pulling hills and mountains out of it.

Noob Note: If you don't want to have to delete the default cube every time you open a new file, just delete the cube once so you have the blank window (only lamp and camera left) . Then select *File - Save default settings* or hit *Ctrl+U*. From now on, each new project you create will start off without the cube.

139.2. Molding the Mountains

The key to making good mountains is using the proportional edit mode (**OKEY**) and constantly adjusting the radius of influence. If you've already gone through the Mountains Out Of Molehills¹ tutorial then this section will be familiar. One major difference is that in this tutorial I recommend rotating the 3D view around so you have a good view of all three axes instead of working in the front or sides view. Using the proportional editing tool affects multiple vertices, and it helps to see what effect your changes are having as you make them.

- Go ahead and turn on proportional editing, either by pressing **OKEY** or clicking on the grey ring on the 3D View header. You have to be in Edit Mode to select this option. Once proportional editing is enabled, the ring will appear orange and a new drop-down menu will appear next to it with different falloff styles. Select Smooth if it is not already selected.



Figure 934

¹ http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Mountains_Out_Of_Molehills

- Select any random vertex and grab it (**GKEY**). You will see a ring around the vertex you are grabbing. This is the radius of influence, and only vertices inside this ring are affected by the transformation. If you are doing this in the orthographic view from the front, side, or top, then this will be obvious. But if you're at a view where you can see all three axes, then it may be less obvious.

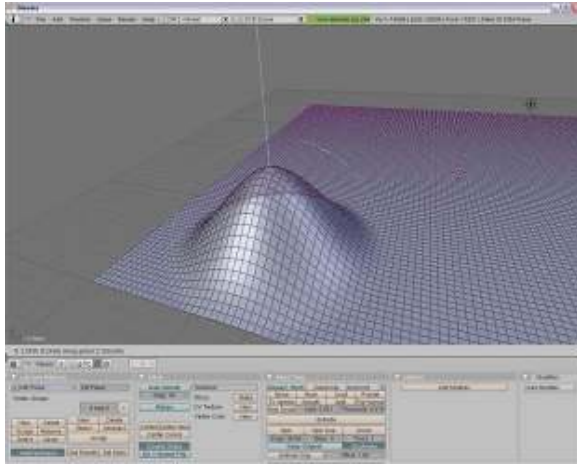


Figure 935 The first mountain created using proportional editing.

- Restrict movement to the z-axis (**ZKEY**) and translate the vertex upward. Throughout this entire process you **ONLY** want to translate along the z-axis. If you start moving vertices in the x or y directions, things become distorted and you get some nasty creases. Play around with adjusting the size of the radius of influence (**Mouse Wheel**) to get steeper or flatter hills.

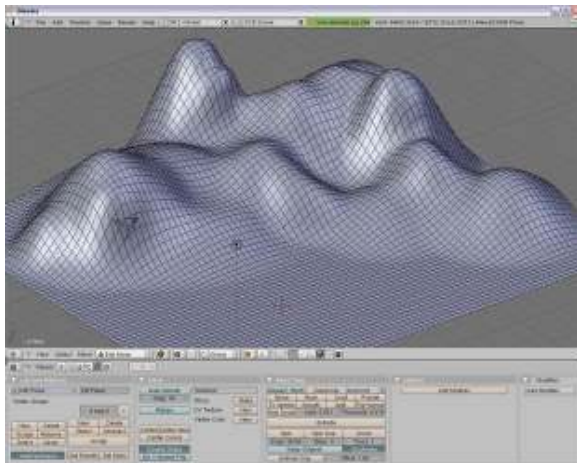


Figure 936 A nice group of hills

- Keep repeating this process with different size radii and different heights until you have a decent mountain range, but leave an area flat. We'll be using that spot later in the second tutorial. Don't be afraid to occasionally translate some vertices down instead of up to create depressions in the hills. Remember, variety is the spice of life. Very few things in nature are naturally geometric, so mix up

your hills and especially make sure they overlap. How often do you see a nice, smooth hill all by itself in nature?

Note: You'll notice in my screenshot that I have reduced the size of my grid. For simplicity's sake, I didn't feel like filling an entire 100x100 grid with mountains since this can take some time.

- Well now, that's looking pretty good! Now, there's one problem with our hills so far. They're too smooth! Let's bumpify them a little. Change the falloff type from Smooth to Random.



Figure 937

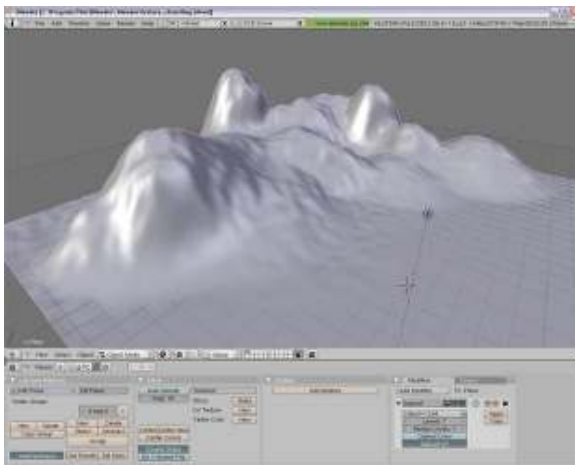


Figure 938 Rougher hills look more realistic

- Select a single vertex and grab it (**GKEY**). We're still working on the z-axis only, to restrict your movement with the **ZKEY**. Now when you move the vertex up and down, all vertices in the radius of influence will also move but with a random falloff instead of smoothly. It only takes a little movement to get the effect we want, so something around 0.5-1.0 is enough. Mix up moving up and down with different vertices, again to add variety to the scene.
- Once you have your landscape the way you like it, add a subsurf modifier under the Editing tab (**F9**) and select Catmull-Clark. This will smooth out your terrain a little so that it's not too rough. Given the number of vertices you already have, it's not necessary to have a higher render value

than 1 unless you just REALLY want it to be smooth, but I don't recommend it. Land is supposed to be bumpy and rocky, we just don't want sharp edges.

Noob note: If you're making mountains using Random Falloff and the peaks stick up too much: in Edit mode, select the points in the area around the peak using circle select, then press **WKEY** and click 'Smooth' until you're satisfied (or, in the Editing tab (**F9**), click the 'Smooth' button).

Useful Tip Proportional editing can be used on multiple vertices simultaneously. This is especially useful if you're trying to create a cliff face or a river bed. Use the box tool (**BKEY**) to select a group of vertices and then translate them. Keep in mind that the size of the radius of influence determines how many vertices around **EACH VERTEX** will be influenced. So suppose you have a 5 vertex radius, that means that 5 vertices all the way around your selected region will be influenced.

139.3. Texturing the Terrain

Alright! We've got some pretty nice hills now! But there's still a few problems. Hills shouldn't be white, and hills shouldn't be SHINY! Let's dress them up a little, shall we?

- With your landscape selected, go to the Shading Panel (**F5**) and add a new material.
- Under the Shaders tab, drop the Specular value to 0.
- Go to the Texture tab (**F6**) and add a new texture.
- In the Texture Type drop down menu, select Image.
- Two new tabs will appear. In the Image tab, click Load and load a texture from file.
- In the Map Image tab, increase the Xrepeat and Yrepeat. Depending on the size of your terrain and the image that you use (please use something that tiles!), these values will vary. I've used 10 for each in this tutorial.

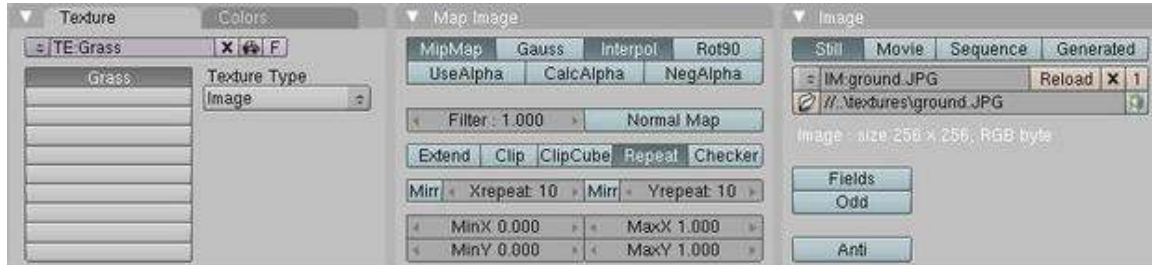


Figure 939

(**Noob note:** Render the scene (**F12**) to see the applied texture) (**Noob note:** A quicker way to do this is by using the render preview tool, 3d view window - render preview, View--> Render Preview or press **SHIFT--> PKEY**) (**Noob note:** you may also select shaded in the Viewport Shading menu next to where you select object mode or edit mode to see your texture on your mountains without rendering, however it does slow your computer down some which could make editing frustrating. So only do this if you want to see what your textures look like, then switch back to solid for more editing.)

- Finally, let's do something about the lighting. Go to Object Mode (**TAB**) if you're not already there and select the lamp. Choose the Shading Panel (**F5**) and then click the icon that looks like a light bulb to display the Lamp buttons. Change the lamp to a sun and up the energy to 1.5. You may

also need to increase the distance if your terrain is large, or rotate it around if you don't like where it's pointing. The dashed line coming out of the light is the direction. Play around with different angles and energy values for the sun to get different times of day in your scene.

And there you have it! Your landscape is now textured with some nice grass to make it look a little more realistic. Play around with hill sizes and roughness if you're not satisfied with your landscape, but try not to be too picky. Nature shouldn't look too sculpted. Remember, you won't notice a change in texture unless you render your image.

NOTE If you went with the suggested 100x100 grid, the rendering process could definitely take some time depending on your system, especially if you have ray tracing and shadows enabled. To boost the rendering speed, go to the Scene panel (**F10**) and under Rendering, disable the buttons that say **Shadow** and **Ray**. Also note that because we've created some hills, your camera may now be under the terrain. Switch to the camera view (**NUM0**) to see what your camera sees and move it if you need to.

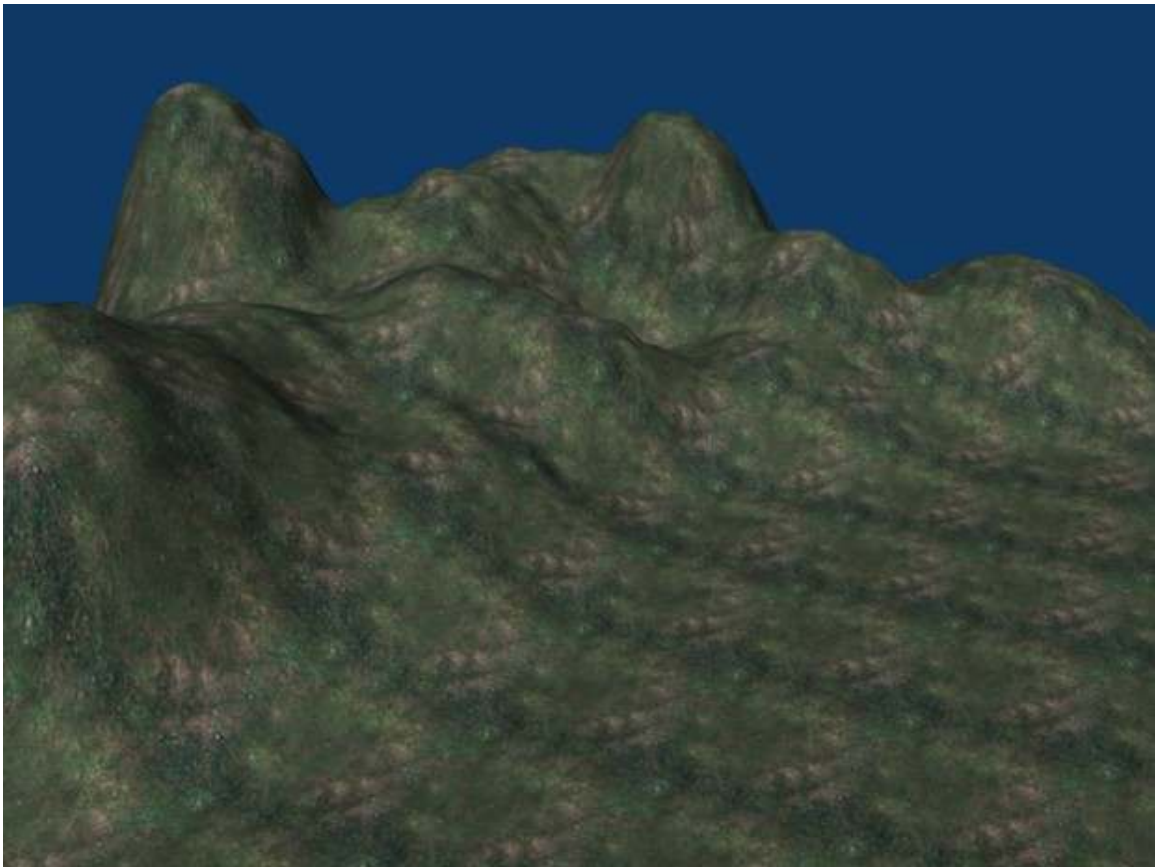


Figure 940

Join us next time as we explore how to make the landscape look even better using texture stenciling! And please, if you have any issues with the tutorial, or feedback (positive or negative), drop it in the discussions. I'll try to keep an eye on it to see what people have to say. - Moohasha

139.4. Reader Contributions

I'm mainly doing this so the hills look better and more realistic. Reading this will consume more time than doing it. I'm making everything clear for beginners. It will only take about 8 minutes more for your hills to end up like this:

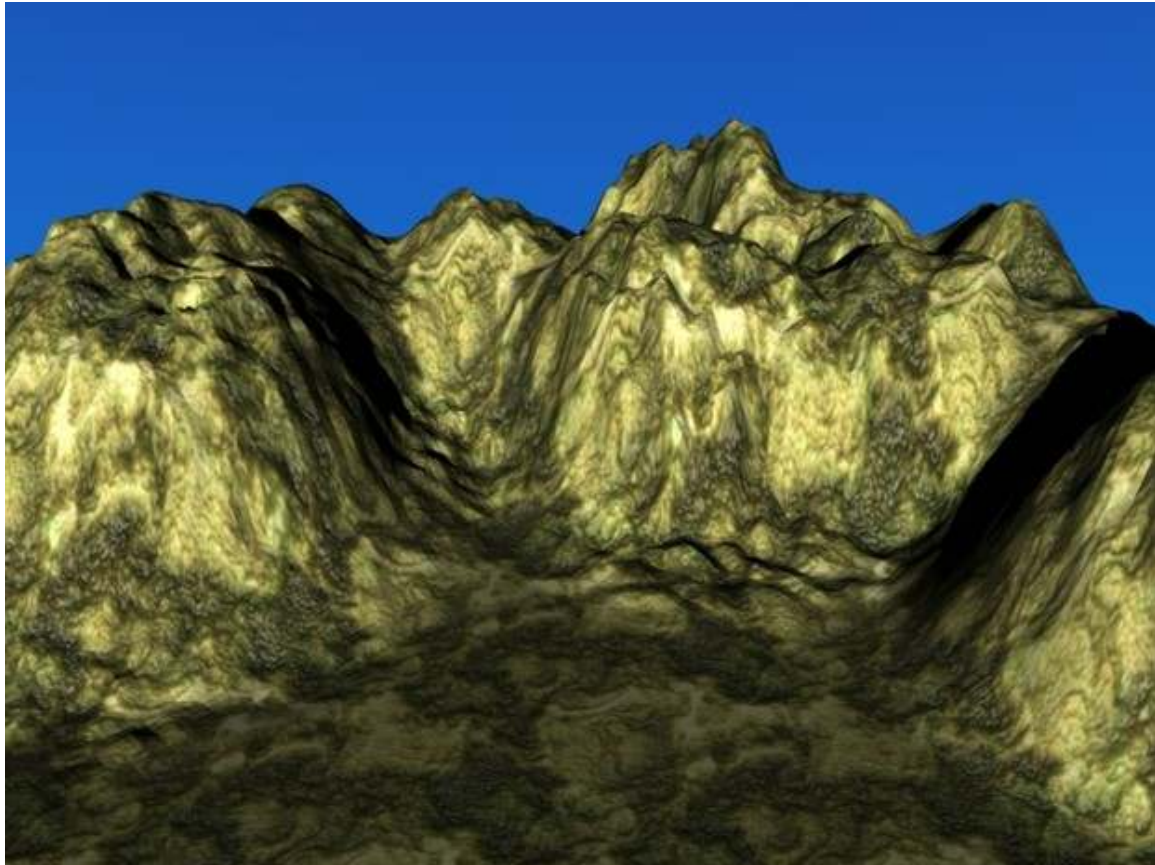


Figure 941

1. First step to achieve it is to switch from the default lamp to Sun.
 - a) You do this by clicking on the default and go to the shading tab (F5)
 - b) Click on *Sun*.
 - c) Go to *rotate manipulator mode* with Ctrl Alt R.
 - d) Rotate the sun until the dotted line is in your desired position. That is where the main energy will go. It will differ if you want to achieve the different time of day. You can make the distance greater or less with *dist.* I kept mine at default 30.
2. Now let's make more realistic, paler sunlight.
 - a) In the RGB slider, set R for 1, G for 1, and B for .848.
 - b) Set energy for 1.63. This will all differ for different times of day, so set your east and west in your head, and the later into the day, the further the sun to the west and the more orange. For midday, keep the energy on 1.63 and put the sun right above your hills. I set mine for earlier in the morning.
3. Now click on the picture of a globe to change the background.

4. Click on the *blend* button for a more realistic looking sky. On the left of the World toolbar, that will be the color of lower down in the sky and the right sliders will be the color of the top of the sky. Naturally, the top should be darker blue than the bottom.
 - a) It will differ for different times of the day. So you might want something different from this. But I set up mine for the morning using the settings below:
 - The left HoRGB sliders to 0.50, 0.68, and 1.
 - The right ZeRGB sliders to 0.11, 0.25, and 0.66.
5. Now for the texture. *This* won't differ for the times of day, but it will differ for what type of landscape you want.
 - a) I got the texture from [here](#)². The best textures to use would be either Desert Canyon, Hill Country, Rough Terrain, Class M Planet, or Red Planet. Use Red Planet for a Rocky, Mars-like look. (Note: these textures are no longer available on this URL as of Nov 2010)
 - b) I chose Hill Country³. Click on it, and I know it looks like snot, click on it again, and save the image.
 - c) Go back to Blender, go to Texture buttons (F6)
 - d) Click add new
 - e) Select *image* for the texture type
 - f) Go to the *image* toolbar, and upload the texture. Blender will only allow you to upload from the Blender documents, unless if you click on the up and down arrows on the top left of the screen and choose the place where you saved the texture (or picture).
 - g) After uploading, the X and Y repeats should be smallish, like 6x6, so the changes aren't noticeable. It will look ugly in the preview, but it will look nice after its wrapped.
6. Render, and 1 minute later, voila! Those are nice looking hills!
7. Now boast in front of your friends!

Noob Note: You need to have your landscape selected if you are in object mode to change the texture, otherwise it will change the texture of the world.

139.5. Reader Contributions 2

You can have an even better result if you use the texture to "bumpmap" the mountains.

1. Press **F5** until you are into Material Buttons
2. Select the *Map to* tab
3. Click on *Nor* once (the *Col* option must stay selected as well)
4. Slide the *Nor* slider to 5 or more (might differ depending on texture size and repeat options)
5. Render with **F12**

2 <http://oneodddude.net/FreeTextures.php?action=browse&catid=5>

3 <http://oneodddude.net/FreeTextures.php?action=display&id=10&page=1&sort=id&catid=5>

139.6. Reader Contributions 3

Alternately from making the hill model manually, you can use a program like L3DT⁴. It uses various algorithms to generate very detailed and realistic terrain heightmaps. It also lets you edit them in a more intuitive way than Blender does. After you have L3DT make your height map, you can export it as a .x file, which you can import into Blender. **Noob Note:** That can be done by going into File->Import->DirectX(.x). And there you have a very realistic landscape mesh with a lot less work. (Noob Question: I want to use Unity3d game engine to create a game, but i want to use blender to create landscapes. To create massive world landscapes, does anybody have any tips? as Unity limits any mesh to 65000 vertices, which i've already passed with my method without doing half the map.)

-If you are using unity 3d just use it for the terrain. Download the terrain toolbox and just use .raw images. Or simply generate from there and save. 65,000 verts is plenty btw. If you insist on using blender for the terrain then model a high poly terrain fist consisting of around 1 mil+ verts. bake the normal map and ambient occlusion decimate your terrain to around 10k or w/e looks good for the game. Then bam you got 10k terrain that looks like 1mil. But to be honest blender terrain texturing is a pain compared to unity. Anyhow hope that helps.

4 <http://www.bundysoft.com/L3DT/downloads/standard.php>

140. Landscape Modeling II: Texture Stenciling

This is a continuation of the previous tutorial, Landscape Modeling I: Basic Terrain¹. In this tutorial, we will make our terrain look even better by using some texture stenciling to add multiple textures where we want them in the landscape. This tutorial assumes that you have a basic understanding of how to use Blender (how to add/remove a mesh, how to change views, etc...)

140.1. Creating the Stencil

The landscape from the previous example looks pretty good, but the entire thing has the same texture, so it doesn't look very natural. Let's add some rocks to those hills. If we just add a second texture to our material, it will completely cover the previous one. What we want is to have the first texture only show in certain places and the second texture cover the rest. In order to do this we'll have to create a stencil, which is like a mask that determines where textures appear on the material. The stencil is a black and white image, similar to a heightmap, except the intensity of each pixel determines how much of the next texture will appear (black = 0%, white = 100%). I highly recommend viewing this² tutorial for a more in depth description of how stencils work. If you think you have the gist, then continue on here. We're going to create our own stencil to determine where we want rock in the landscape by "painting" on the object. ***IMPORTANT*** The next few steps will be performing temporary modifications to your scene, so be sure to save your file before you continue so that the changes will not be saved!! Also, if you have a subsurf modifier on the landscape, remove that at this time as it drastically slows down the following process.

- Select the landscape object and switch from "edit mode" or "object mode" to "**Vertex Paint mode**". This is found in the Mode drop down menu, on the 3D View header. This mode lets you paint the object, and thereby change the color value of each vertex. When in Vertex Paint mode, you'll notice your object change color to a pixelated version of the texture that is applied to it. This is because with that texture applied, each vertex will be drawn with the colors you see.

Noob Note: Vertex Paint mode is in the drop down menu that contains Object mode and Edit mode. Just select the object while in Object mode, click the drop down menu, and select Vertex Paint mode from the list.

- Switch to the Editing menu (**F9**) and you'll see a new tab called Paint. Notice the sliders labeled Opacity and Size. Opacity determines how much to blend the selected color with the existing colors when you paint. So 0.2 means that when you click, you'll get 20% of the selected color mixed with whatever is currently there. Over on the right are different options to combine to colors. We want to mix, so as we paint the colors will mix together and give us a nice smooth blend. Size is the size of your cursor while painting, and thus how many vertices are affected. Keep in mind

¹ Chapter 139 on page 1089

² <http://feeblemind.org/blog/index.php?post/2005/02/27/16-didacticiel-usage-des-masques-de-tex>

that does not change as you zoom in and out, so a size of 10 can actually paint more pixels if you're zoomed way out than a size of 30 if you're zoomed way in.

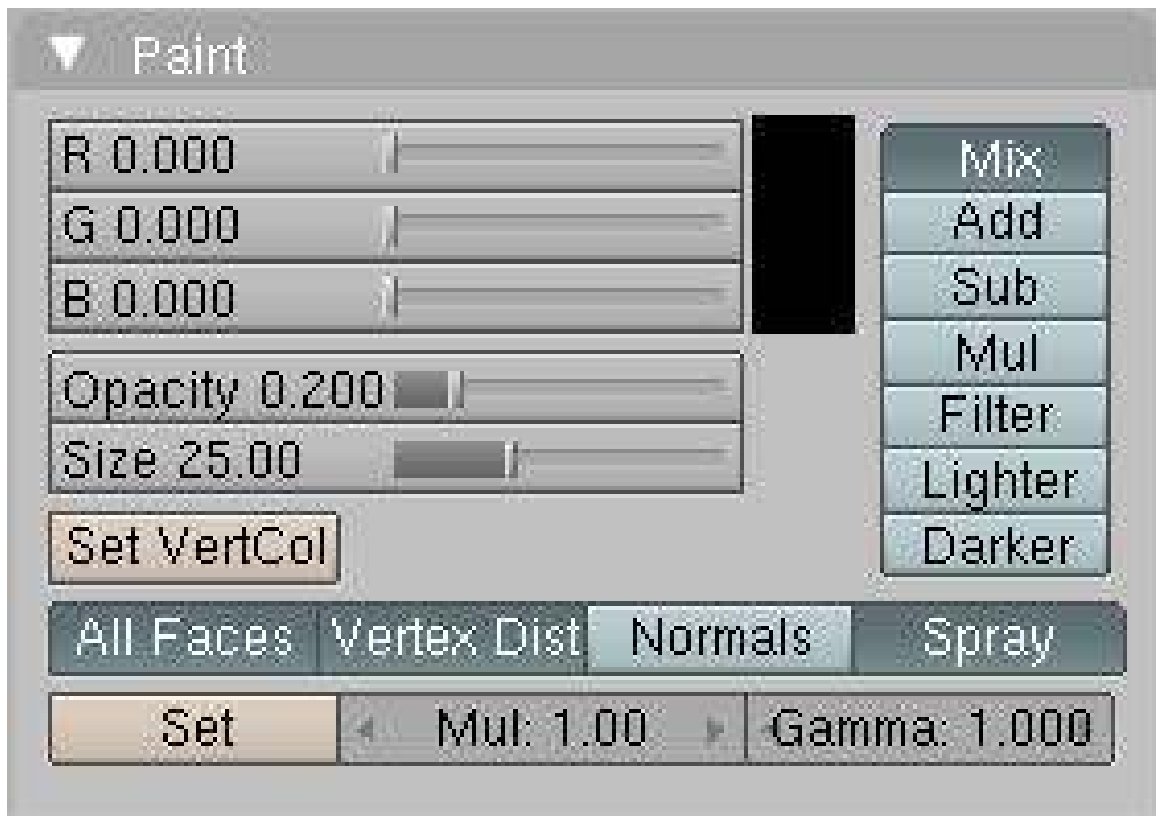


Figure 942

- Change the color to black and click the button that says Set VertCol (in 2.48 version, that button is below the color square and named 'SetVCol'). This will change the color of your entire object to black. You'll notice that the object is not lit in Vertex Paint mode, so when the entire object is black it can be difficult to see where your hills are. You'll have to move the camera around to see the shapes.

[Another newbie recommends: if you switch to the "object" buttons (f7) and turn on "Wire" under "Draw Extra" in the "Draw" panel/tab you should see a wireframe on top of the vertex colors you are painting, which makes it much easier to see where the hills are. Remember to switch this back off when you have finished painting.]

- Now we're going to paint the places where we want rock to show through. Change the paint color to white. It's less likely that grass will grow on steep slopes, so start painting the tops and edges of the steeper hills white by holding down **LMB** and dragging your mouse (think MSPaint). If you didn't heed my earlier advice and still have a subsurf modifier in effect, the painting will be very choppy, so remove that now.
- As you're painting, try to make some spots pure white by going over them again and again, but don't make any one area of white too large as this will make a huge area be all rock, and we're trying to blend two textures.

- This process can just be trial and error getting the landscape painted the way you want. Once we apply our stencil later you may decide there's too much rock in one area, or not enough in another and go back and change it. A few things to keep in mind:
 - 20% opacity means that most of the original texture (grass, in our case) will show through, so the rock won't be very noticeable if at all. You'll have to go over the same spots a few times to increase the intensity.
 - You don't want a dramatic change from grass to rock, so be sure to blend white areas with black. That's why we're using the 20% opacity instead of just bumping it up to 100%.
 - Make your patches random and spotty. This will end up creating a more realistic effect once we combine the textures.

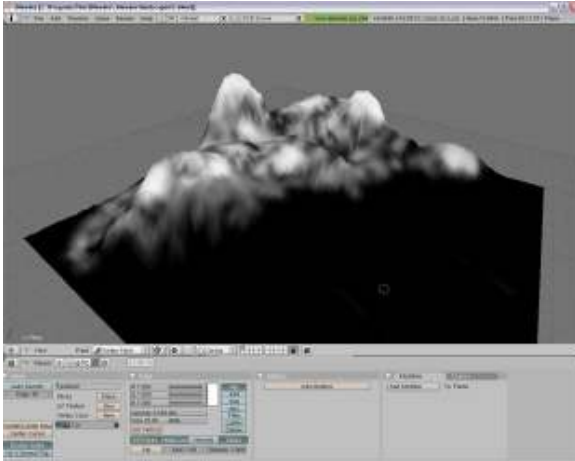


Figure 943 Painting the stencil onto the landscape

- Once you're done you should have something that looks like this. Notice the mixture of white and black in some places, how it doesn't just go from pure white to pure black as you move down the hills. This will make a more natural blend and it will cause random rocky areas mixed in when the grass. Now we need to turn this into our stencil.
- Go to the overhead view (**NUM7**) and make the projection orthographic if it's not already (**NUM5**).
- Zoom in/out (**Mouse Wheel** or **NUM+/NUM-**) until the plane almost occupies the entire window.
- Move the 3D cursor and any lamps or cameras away so that they're not over the landscape and take a screenshot.

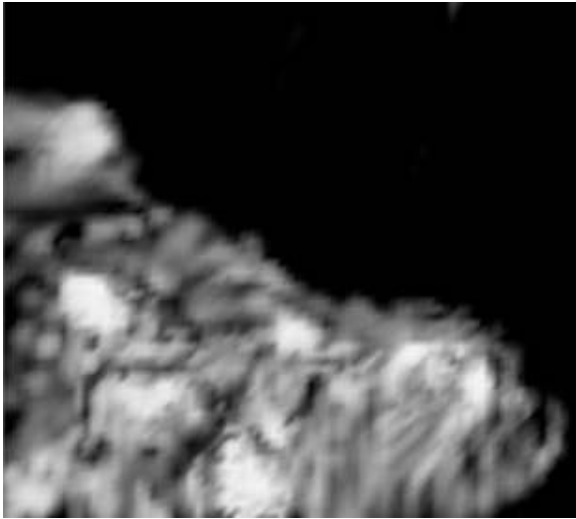


Figure 944 The stencil after its been cut from your screenshot

- Open your favorite image editor (I'm a traditionalist, so I still like MSPaint), and cut out the image of landscape. This is now your stencil. Save it under your favorite format.
- Re-open the saved version of your landscape to undo the changes we made to create the stencil.

So why did we make our stencil this way? This allowed us to actually paint on our terrain so that we ensure we get the rock exactly where we want it. Using this method, you can customize your stencil to any object, as long as it has enough vertices. For example, if we just had a flat plane made of 4 vertices, this technique would not have worked because we could only paint the corners. That's why in the previous tutorial I recommended using a high (but not too high) number of vertices in your grid.

140.2. Applying the Stencil

So now we need to apply the stencil to the landscape to mix our grass and rock textures together.

- Return to either Object Mode or Edit Mode (**TAB**), it doesn't matter which.
- Go to the Texture menu (**F6**) and add two new textures in the material (you should already have the grass texture).
- Make texture 2 (the first new one you added) an image and load your stencil from file. Do not repeat this in the x or y directions. We want it to map to the entire object.
- Make texture 3 an image and load your rock texture from file. This one you do want to repeat, just as you did the grass.
- Now return to the Materials menu and select your stencil texture. It's a good idea to name your textures, materials, objectives, etc... so that they're easily identifiable.
- With your stencil texture selected, switch to the tab labeled MapTo. Deselect Col (which maps the texture to the color), and select Stencil and No RGB. Stencil will treat this image as a stencil, and No RGB treats it as a black and white image. If you didn't select this second option your stencil wouldn't work.

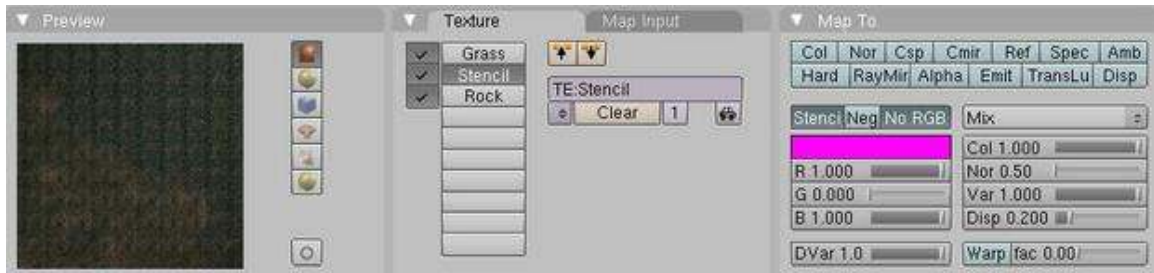


Figure 945 Mapping the stencil texture

- In the preview tab you should notice your material change so that the grass and rock are now mixed. If you render, you'll see a much more realistic landscape than the one from the previous tutorial!

Noob Info: Be sure that for the rock - texture *col* is selectet because we want to color the landscape with the rock-texture! Otherwise it doesn't work. *Select Rock-Texture -> Map To -> col*

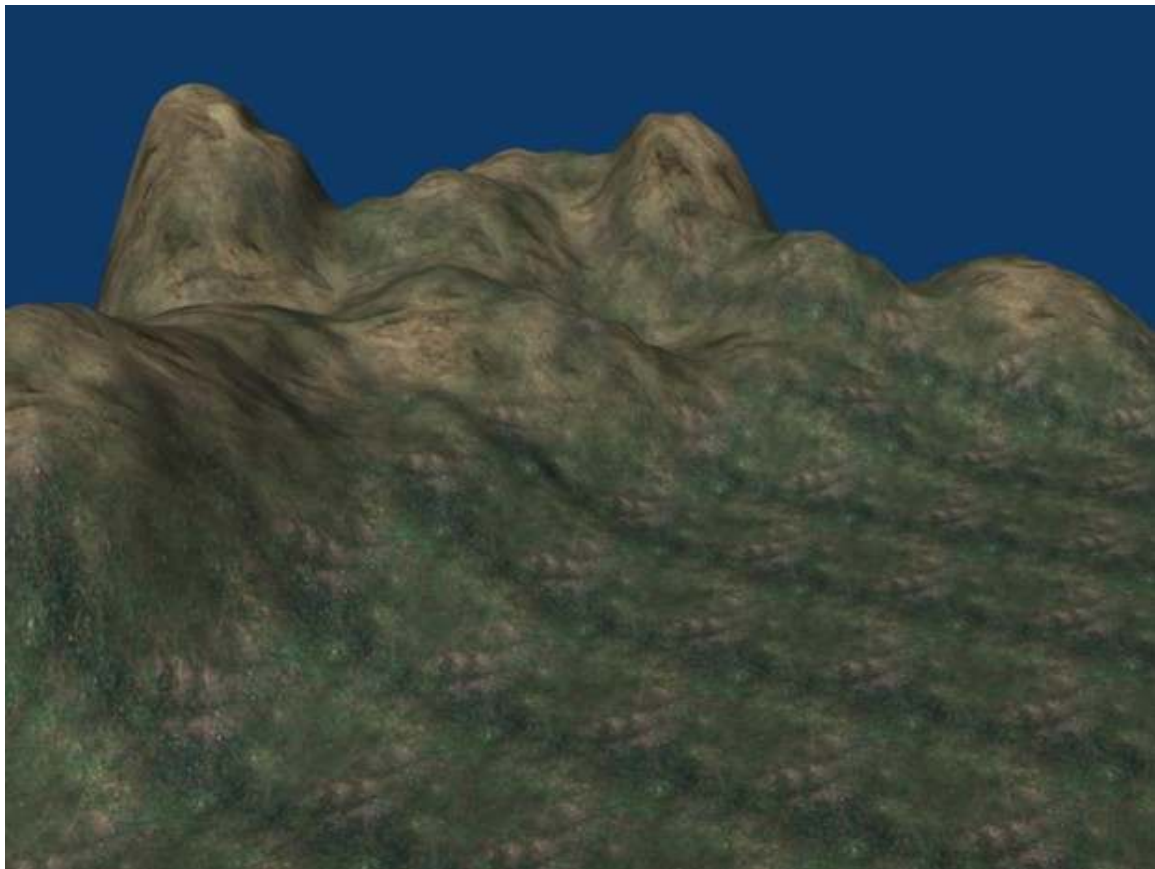


Figure 946 Landscape with grass and rock textures mixed

140.3. Adding Snow

Since you already have some nice mountains, why not add some snow to them? Adding snow is easy. All you have to do is add a fourth texture AFTER all the others. Make this texture the same as your stencil.

- Make sure you're in Shading (F5) > Texture Buttons (F6) > Textures.
- Click on an empty panel beside Texture Type
- Click on the little box to the left of Add New
- Choose your stencil from the menu (hint: it helps if you name your textures)
- Render and see the results.

If you made your stencil properly, you should have some nice, snowcapped peaks. It's not much, but it's enough for a quick fix. If it isn't even, you probably didn't make your stencil properly (meaning pretty much all white on the peaks, and not much anywhere else). Of course, there are better ways to do this, ways I wouldn't know about; I discovered Blender merely three days before the time of writing. I leave it up to you to see the effect: I only have so much room and bandwidth for pictures!

This effect works by saturating the underlying color with white, depending on the amount of white in your stencil. A light gray will saturate it only a bit, whereas a bright, 100% white will cover it very clearly and efficiently. If you look closely, you'll see that the light gray on the stencil DOES make a difference, but it's not immediately visible and requires lots of fidgeting with the Render Preview and texture options. You might also try fiddling with the Stencil and NoRGB option, although it worked fine for me with both turned off (I use Blender 2.48a)

Some challenges:

- Try to change the color of the snow. Hint: RGB
- I've yet to do this, but try making snow out of a heightmap.

140.4. Multiple Stencils

Remember in the first tutorial when I said to leave part of the landscape flat, because I'd be using it later? Well it's time to use it. I want to eventually use the landscape as the backdrop for a military base, so let's texture the ground around where the base will go to give it a dirt ground and road leading to it. ***IMPORTANT*** Just as before, the next few steps will be performing temporary modifications to your scene, so be sure to save your file before you continue so that the changes will not be saved!!

- Switch to the Vertex Paint mode in the mode menu on the 3D View header.
- Go to an overhead view (**NUM7**) and make the projection orthographic (**NUM5**)
- Once again, paint the entire object black by selecting the color black in the Paint tab and clicking on Set VertCol.
- For the next few steps, you may need to **TAB** back and forth from Vertex Paint mode to Object mode to make sure you're painting in the correct area. Make the opacity of the painter 1.0 and the size 10 so that we can work very precisely.



Figure 947 The stencil for our future base

- Paint an area over the flat terrain in the shape of a generic military base, like you see here.
- Reduce the opacity back down to 0.2 and increase the size to 20.
- Paint the road leading away from the base. We've reduced the opacity and increased the size so that the road will blend more with the landscape around it instead of being more defined like our base.
- Once you have your stencil looking the way you want, take a screen shot and save it. Then revert back to your previous environment (before you started painting on it).
- Go to the Texture menu (**F6**) and add two new textures.
- Load your stencil into the fourth texture (the first new one), and the dirt image into the fifth texture. Once again, remember to set the x and y repeat for the dirt texture.
- In the Materials menu (**F5**) select your new stencil and go to the MapTo tab. Deselect Col again, and select Stencil and No RGB. Notice the new preview.
- Wait a second, where's the dirt path?!? Here's the problem and the reason I've included this section: stencils are cumulative. That is, the first stencil defines which portions of ALL successive textures will be drawn, including other stencils! In order for our new dirt path to show up, we need to make sure it's part of the first stencil.

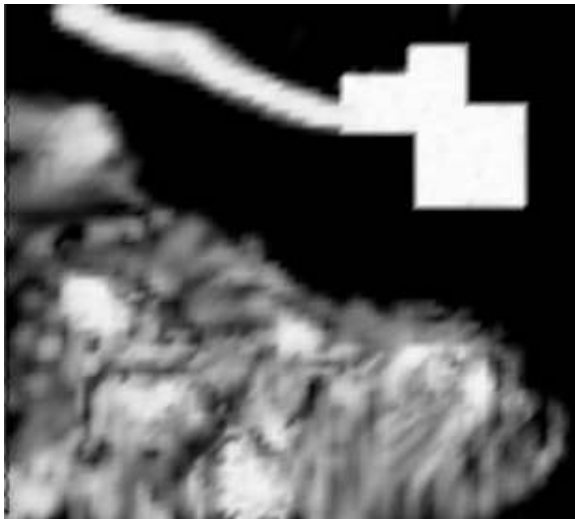


Figure 948 Both stencils combined into one

- Open your first stencil (the one you used to add rocks to the hills) in your image editor and combine the new stencil with it. The result should appear somewhat like this. Be sure that when you combine your dirt stencil with your rock stencil that the dirt stencil remains the same size and in the same place. That is, you want the two to be perfectly aligned.
- Reload your first stencil on your material, the one we used for the rocks. Now the preview should look the way we want it to, and if you render you will see your new dirt area on the flat part of the landscape. You can add even more stencils the same way. Here is how all of the textures blend together:

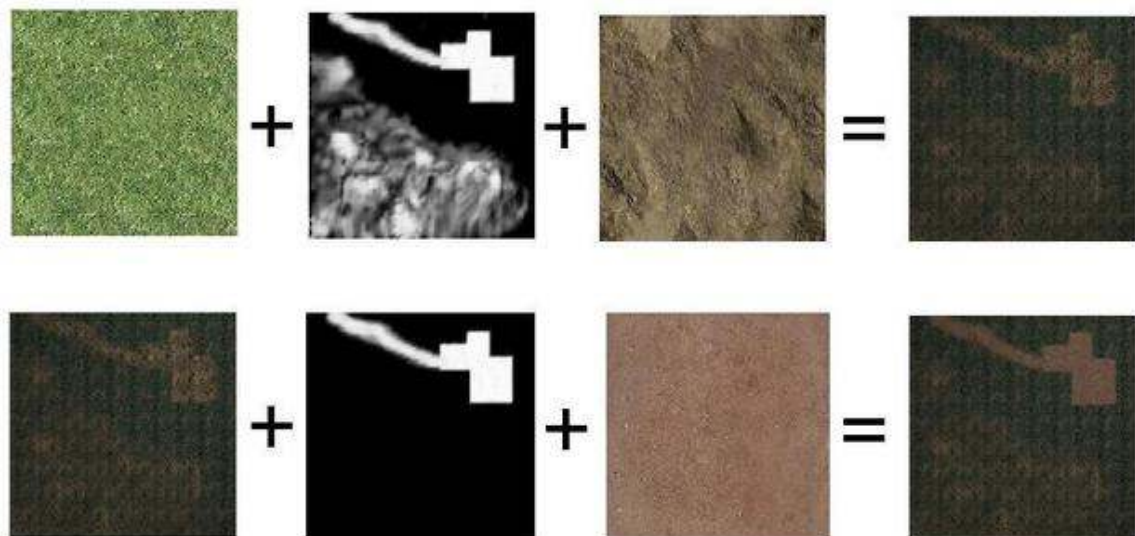


Figure 949

NOTE I had to replace the grass texture in the above image since I could not find the licensing information for it. I apologize for the inconsistencies in the diagram that result from this.

Well there you have it! This is a pretty useful method of texturing complex objects. Check out our new landscape and how it compares to the previous, boring, single-textured landscape!



Figure 950

Once again, if you have any issues with the tutorial, or feedback (positive or negative), drop it in the discussions.

141. Landscape Modeling III: Exporting as a Heightmap

In this tutorial, I will show you how to export your beautiful terrain to a heightmap that can be used in most 3D graphics engines. The benefit to doing this as opposed to just exporting your mesh is that many engines have a special process for dealing with terrains as opposed to regular meshes where it divides the terrain up into different sections so that it can subdivide regions closer to the camera more for greater detail, and also cull (not render) regions outside the camera's field of view, saving precious processing time. This structure is called an oct tree¹ and is a highly optimized way to render large meshes such as terrains.

Before I begin, I must give credit to the [Creating a Heightmap from a Plane](http://wiki.blender.org/index.php/Tutorials/Creating_a_Heightmap_from_a_Plane)² tutorial on the Blender wiki³. This is how I learned to do this trick, and most of what I will cover came from this tutorial. I'm simply including it here with the other landscaping tutorials I've written for convenience.

141.1. A Word About Heightmaps

I will briefly cover what a heightmap is and how it's useful for those who may not know. If you are familiar with heightmaps, you can skip ahead to the next section.

1 http://en.wikipedia.org/wiki/Oct_tree

2 http://wiki.blender.org/index.php/Tutorials/Creating_a_Heightmap_from_a_Plane

3 http://wiki.blender.org/index.php/Main_Page

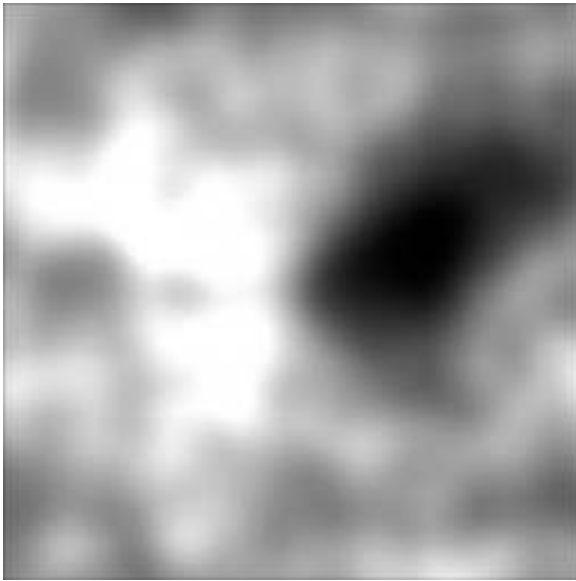


Figure 951 An example of a heightmap

A heightmap is a grayscale image that uses various shades of gray to represent different elevations across a map. Since the images are 8-bit (with the exception of some RAW formats that are 16-bit), you have 256 different shades of gray, ranging from pure black (0) to pure white (255). Black represents the lowest elevation on the map, while white represents the highest. Many 3D graphics engines already have functions to read in a heightmap and generate terrain from it. The way it does this is it creates a grid of vertices (like we are going to do in Blender in just a moment), and uses the heightmap to determine the elevation of each point on the grid. The more intense the pixel color (the closer it is to white), the higher the elevation of that vertex. In most cases, if the resolution of the heightmap is smaller than the resolution of the terrain, the engine will interpolate the vertices in between those set by the heightmap. For example, if your heightmap is 256x256 and your terrain is 1024x1024, your heightmap will determine the elevation of every fourth vertex, and the three in between will be interpolated.

Note that these values do not represent the absolute height of any given pixel, but rather the height relative to the rest of the map. That is, your heightmap does not represent a landscape with altitudes ranging from 0 to 256. When importing a heightmap, you can specify the minimum and maximum altitudes for the map, and the whole thing gets scaled. For example, suppose you decide the minimum is 1000 feet, and the maximum is 5000 feet. When the terrain is rendered, black pixels will represent 1000, white pixels will represent 5000, and a pixel that is exactly in between black and white (128) will represent 3000 feet (half of the sum of min and max). This makes heightmaps relatively flexible.

Unfortunately, as you may have already noticed, there is a limitation of using heightmaps. Since you only have 256 shades of gray, you only have 256 possible elevations. This causes a problem. No matter how precise your vertices may be in the mesh that you model, no matter how smooth you may have it looking, each point will get rounded off to an integer value between 0 and 255. When used to render a terrain, this can cause a "stair step" effect as the terrain goes from one discrete elevation to the next instead of making a smooth transition. But there are ways to avoid this, which I will cover later.

I apologize if you're scratching your head right now saying "Huh?" For a further explanation and example, check the Wikipedia page on heightmaps⁴.

141.2. Creating the Material

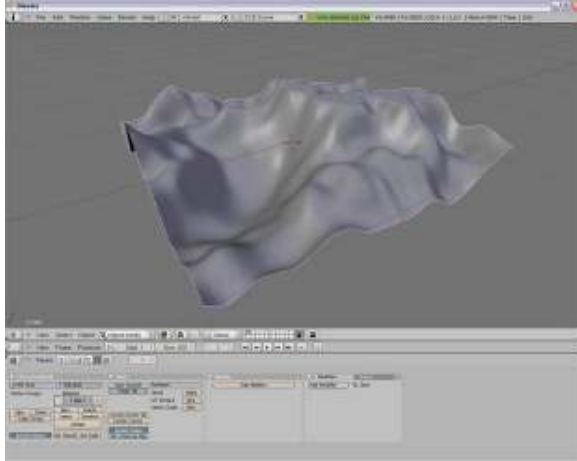


Figure 952 The terrain that will generate the heightmap

To begin this tutorial, I will assume that you already have a landscape that you want to create a heightmap for. If not, read my first tutorial, Landscape Modeling I: Basic Terrain⁵, to quickly create something. **WORD OF CAUTION!!!** If you've already created a landscape, try to remember the exact size of the plane that you used. If you're about to make one, it's okay to scale the plane to have a larger surface to work with, but REMEMBER how much you scale it by. This will make things easier later. I recommend just working with the default 2x2 plane or grid and zooming in on it. You may also want to backup your scene since we'll be changing a few things

To create the material we need, do the following steps:

- Remove any existing materials that may be applied to the terrain.

Note: to remove a material, I think you need to go to the links and pipeline window in the shader section and click the little "X" underneath the words "link to Object"

- Add a new material in the Shading tab (**F5**).
- Go to the Textures tab (**F6**) and add a new texture. This should be the only texture on your material.
- From the Texture Type drop down menu, select Blend.
- In the same window, find the Colors tab and select Colorband. This will let us define the blend pattern, which by default fades from black with full transparency to a solid cyan. Why cyan? I have no idea. It's an odd default.
- Select the black color by clicking on the left side of the colorband. You should see a little black and white bar selected.

⁴ <http://en.wikipedia.org/wiki/Heightmap>

⁵ Chapter 139 on page 1089

- Increase the Alpha all the way up to 1 by adjusting the slider labeled "A".
- Select the cyan color and change this to a solid white.
- What you have now should look like my colorband below.



Figure 953 The blend colorband

- Return to the Materials tab (the red ball icon next to the cheese looking icon).
- In the Material pane, select the Shadeless option to disable lighting on the material.
- To the right, find the pane labeled Map Input. In the bottom of this pane is a grid that by default reads:

XYZ

XYZ

XYZ

- Set all three rows to **Z** so that the texture will map entirely to the Z coordinate.
- In the preview window to the left, if you select the sphere or cube you should see that it goes from black on the bottom to white on the top. That's the effect we want for the terrain. Remember, in a heightmap, black represents the lowest point and white the highest.

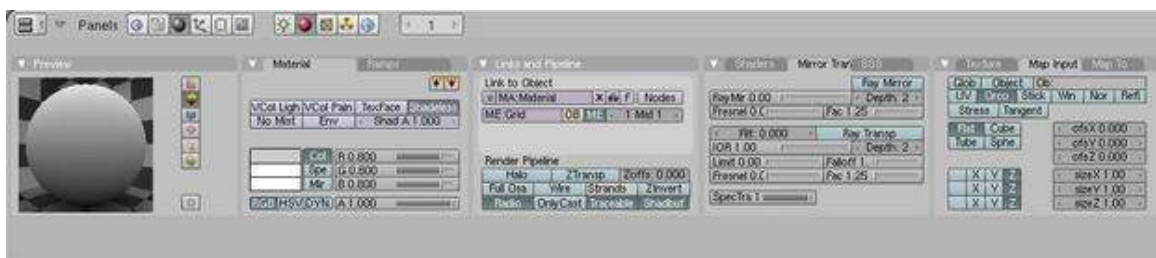


Figure 954 Map the input to the z-axis and make the material Shadeless

- Finally, switch back to the Editing tab (**F9**) and click the Set Smooth button to enable Gouraud shading⁶ so that everything appears smooth.

⁶ http://en.wikipedia.org/wiki/Gouraud_shading

141.3. Setting up the Camera

That's all we need to do for the terrain itself. Now what we're going to do is setup the camera in such a way that when we render the scene we'll get an orthographic projection straight down on the terrain so that we can save it as our heightmap.

- If you still have any lights or cameras in the scene, delete them.
- Switch to the top view (**NUM7**) and place the 3D cursor at the origin (0, 0, 0) by left clicking near the origin, hitting **SHIFT+SKEY** and selecting *Cursor to Grid*.
- Add a new camera, *Space->Add->Camera*. This will add a new camera to the scene that is looking straight down.

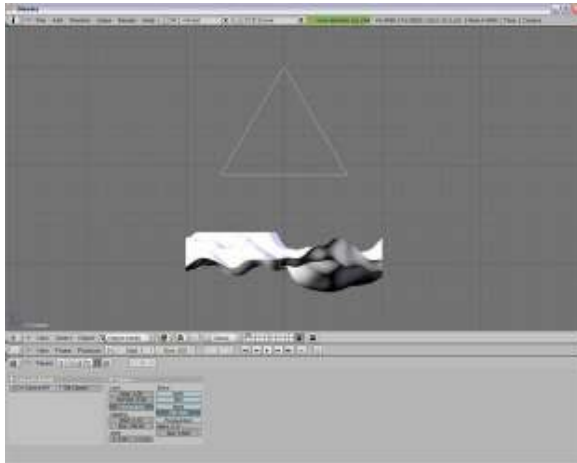


Figure 955 The camera needs to be above the terrain

- Switch to a side view (**NUM1** or **NUM3**) and move the camera up the Z axis so that it is above the highest point in your terrain. Even though the view is going to be orthographic, and therefore distance doesn't matter, the entire terrain still needs to be in front of the camera.
- Now, with the camera selected, go to the Editing tab (**F9**) and select the "Orthographic" button. This will change the camera to an orthographic projection which basically ignores the z-coordinate for all vertices (except for determining which pixels should be in front of others).
- Just above the orthographic button is a value labeled "Scale". Remember earlier when I told you to remember how big your terrain is? This is why.
- Change the Scale of the lens (default: 6) to match the size of your terrain. If you left the plane the default size, this should be 2. If you scaled the plane up by a factor of 20, as suggested in the "Landscape Modeling I: Basic Terrain" tutorial, this should be 40 (the plane is initially 2 units square).
- Now go to the scene tab (**F10**) and under Format, change the SizeX and SizeY to be the size you want your heightmap to be. For most graphics engines, this is required to be either a power of 2 (2^n) or one more than a power of 2 (2^{n+1}). So try something like 256 (or 257 if you need it to be 2^{n+1}).

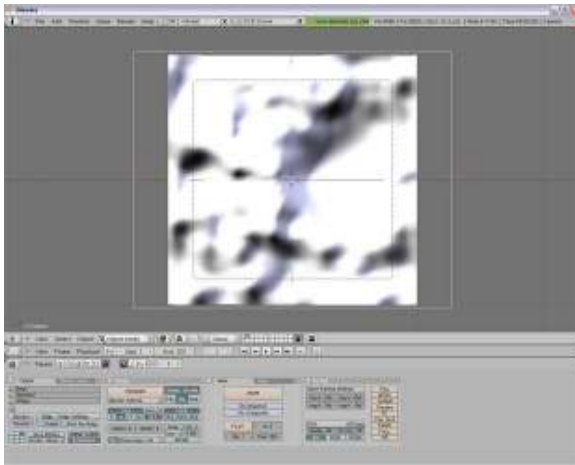


Figure 956 What the terrain will look like from the camera view

- Now, if you switch to the camera view (**NUM0**) you should see a "flat" square that completely fills the dotted box which represents the camera's field of view.

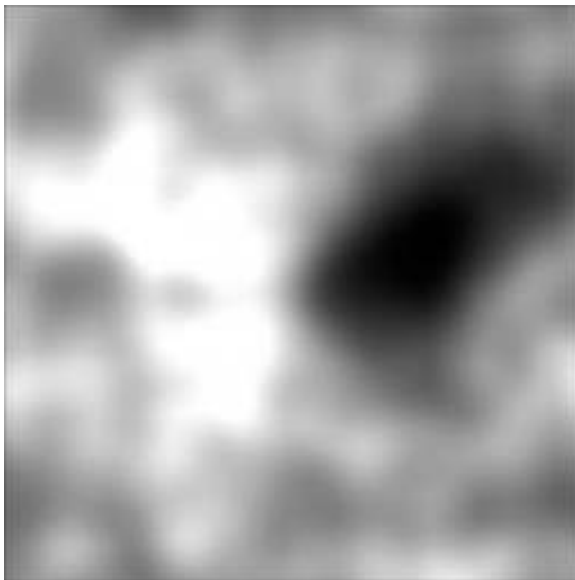


Figure 957 The rendered image will be your heightmap.

- Render the scene (**F12**). You should be looking at your heightmap. Hit **F3** to save the rendered image and you're good to go!

Since this image has no color, and some engines require that your heightmap be a grayscale image (only one color channel instead of three), select the **BW** button in the Format pane before rendering. This will render it as a grayscale image instead of a color image.

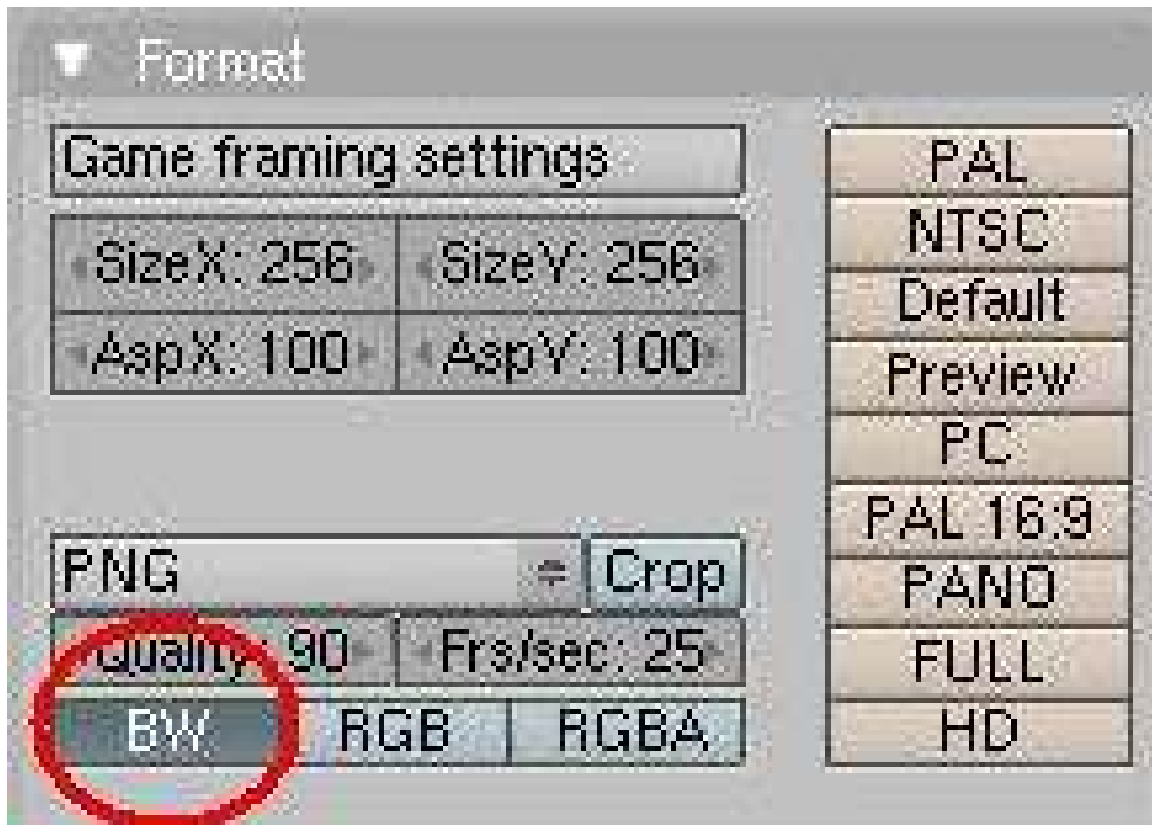


Figure 958 Select BW to save as a grayscale image.

141.4. Avoid Stair-Stepping

As mentioned earlier, exporting the heightmap as an 8-bit image (Blender doesn't support any 16-bit formats that I'm aware of) has the drawback that you only have 256 discrete height values which can cause a stair-step effect when the terrain is subdivided. Ideally, you could use an application that can export the image in a 16-bit raw format, such as Terragen. But this is a Blender tutorial, so we'll not go into that. =P

141.4.1. Reducing the resolution

One way you can "smooth" out the rendered terrain is to actually reduce the resolution of the heightmap. This may sound counter intuitive, but bear with me. Suppose you have a heightmap that is a simple blend from black to white going from left to right. If your heightmap is 1024x1024, you will get 4 columns of each height since you only have 256 different values. In other words, your grayscale will look something like this:

```
00001111222233334444...
```

```
00001111222233334444...
```

```
00001111222233334444...
```

00001111222233334444...

00001111222233334444...

00001111222233334444...

If you try to render terrain from this, you'll get 4 points with 0 altitude, 4 with 1, 4 with 2, and so on, creating the stair-step effect.

Now suppose you make the same heightmap, but reduce its size to 256x256. Now you'll have 1 column for each height, and your grayscale will look something like this:

012345678...

012345678...

012345678...

012345678...

You may be thinking, "But then my terrain will be smaller, or less detailed." But it turns out the opposite is actually true. Consider applying the above heightmap to a terrain that is 1024x1024. The values from your heightmap will be applied to every 4th vertex, and the ones in between will be interpolated. So instead of having something like:

0.0 0.0 0.0 0.0 1.0 1.0 1.0 1.0 2.0

which is what you would get with the 1024x1024 heightmap, you'll actually get something like this:

0.0 0.25 0.5 0.75 1.0 1.25 1.5 1.75 2.0

Where the integers are the values from your heightmap and the fractions are interpolated values. This is assuming that whatever engine you are using will interpolate the values between heights read in from the heightmap.

At the present time, this is the only method I have verified that helps reduce the effects of stair-stepping. As I try different techniques I will update this page with them. For example, I know that Adobe Photoshop can export images in 16-bit raw formats, but I haven't tried it yet so I don't want to suggest it as a solution.

As always, I do these tutorials for you, the reader. So if you have any issues with the tutorial, or feedback (positive or negative), drop it in the discussions. I'll try to keep an eye on it to see what people have to say and update accordingly. Moohasha⁷ (talk⁸) 19:02, 17 December 2007 (UTC)

⁷ <http://en.wikibooks.org/wiki/User%3AMoohasha>

⁸ <http://en.wikibooks.org/wiki/User%20talk%3AMoohasha>

142. Realistic Water using Fluid Sim and Yafray

(This was written using Blender v2.43)

142.1. Using the Fluid Simulator and Yafray to create realistic looking water

This tutorial will (hopefully) successfully walk you through creating semi-realistic looking water using the Fluid Simulator in Blender, then rendering in Yafray.

142.2. Getting Started

When you first open blender there will be nothing but a blank cube.

1. Make sure you are in object mode, and scale the cube up to about 5. Do this by pressing S in the 3d view and holding ctrl, move your mouse until the numbers in the bottom left of the screen are 5. Or select the object and press S then 5.



Figure 959 Where the 5's should appear

1. Next, in the viewport shading box select wireframe. This lets you see through the cube.

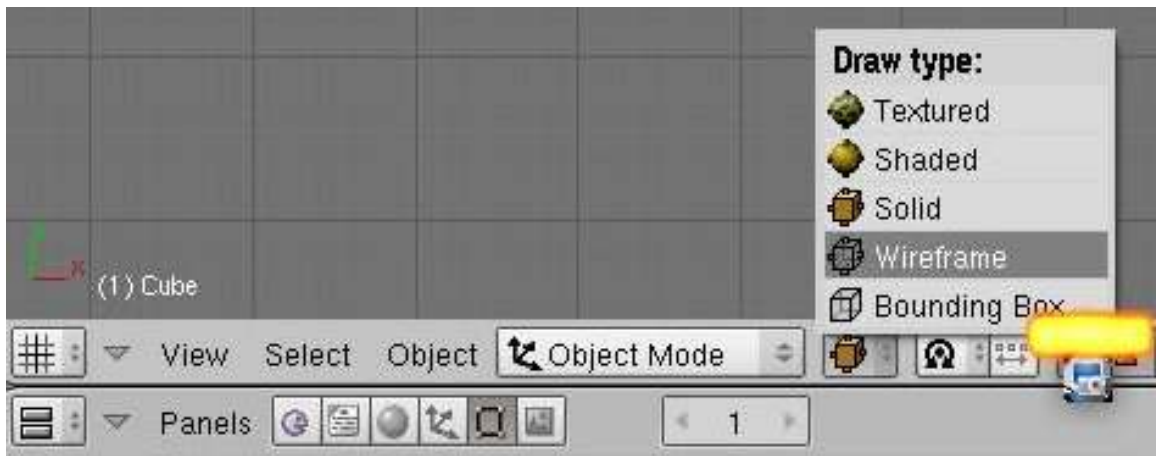


Figure 960 Where the wireframe box is (ignore red and blue thing)

1. We are now going to enable the cube for Fluid Simulation. With the cube still selected in object mode, press F7 (object button) and then press the "Physics Button".

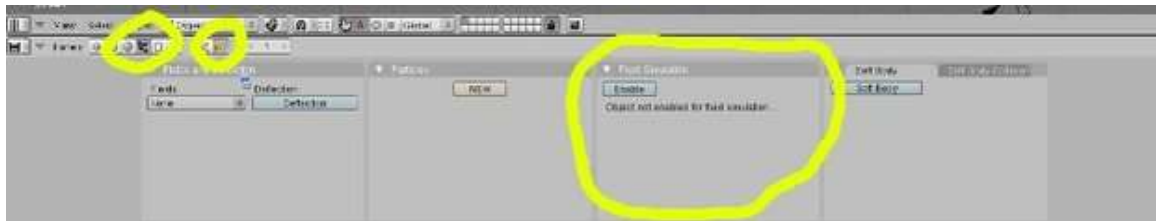


Figure 961 Where the fluid sim tab is

1. You should see a tab labeled "Fluid Simulation" press Enable and then Domain. After pressing Domain several buttons will pop up, these will be used later.



Figure 962 The fluid sim tab

1. Make sure you are in top view now, if not just press NUMPAD7. Next press space>Add>Mesh>Cylinder. 32 vertices is ok.
2. Now middle click and drag in the 3d view to see the cylinder inside your beautiful box. (yay!) What's going to happen is the cylinder is going to be turned into liquid, and the big box is where the liquid can flow about. (it is the "invisible wall")
3. Move the cylinder to a corner of the cube, you can give it some height if you like. Now with the cylinder still selected press Shift-D to duplicate it. Move the new cylinder to the opposite corner.

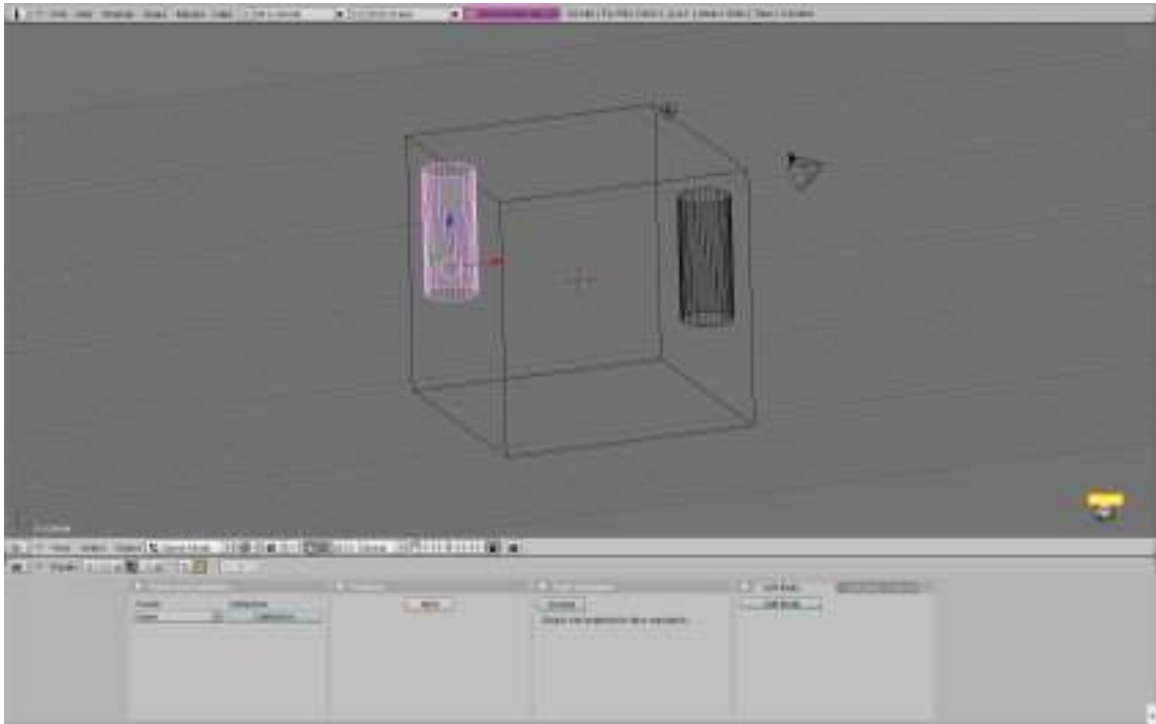


Figure 963 What you should have now

1. On both of the cylinders go the fluid sim. tab and press Enable then Fluid. This tells blender that these cylinders are fluid.
2. **IMPORTANT!!!** After this, you **MUST** subsurf both of the cylinders, but don't apply it. Instead, when the cylinder shows a preview of the subsurf, click the "X" in the top-right of the subsurf preferences.

(noobie note: I don't know what the above step is for, adding the subsurf then click "X" removed it.)

1. Switch to Object Mode then Right click on the big box (the domain) and press the big BAKE button. A little box will pop up and just click on the top option.
2. Now you should see a blob looking thing appear in the cylinders, and it will fall to the ground and splash around. Your computer is "baking" the liquid. (Calculating where it should go)

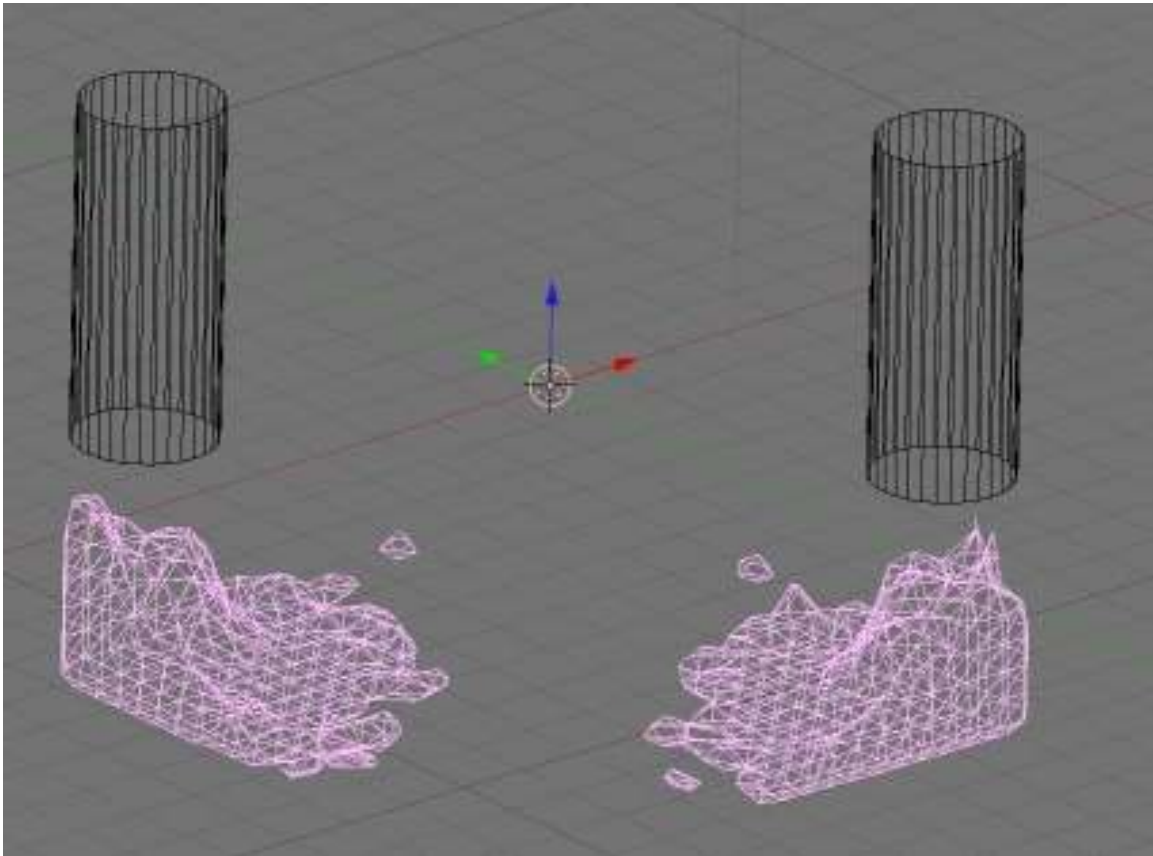


Figure 964 What you should have now

1. As you see now, the liquid isn't very realistic looking. If you do a render, you will see. You would also see that the cylinders are appearing in the image. This is not wanted, so with both of the cylinders selected, press **M** then click on one of the little boxes that isn't already pressed. This moves them to another layer.
2. Now we are going to move the camera to a different position for a better view. Move the viewport around until it's where you think will give you the best view and press **SHIFT-NUM0** to make the camera match your current view.
3. Drag the newly placed camera back a bit, so the when **NUMPAD0** is pressed most of the scene is in view. You should now have something that looks similar to this:

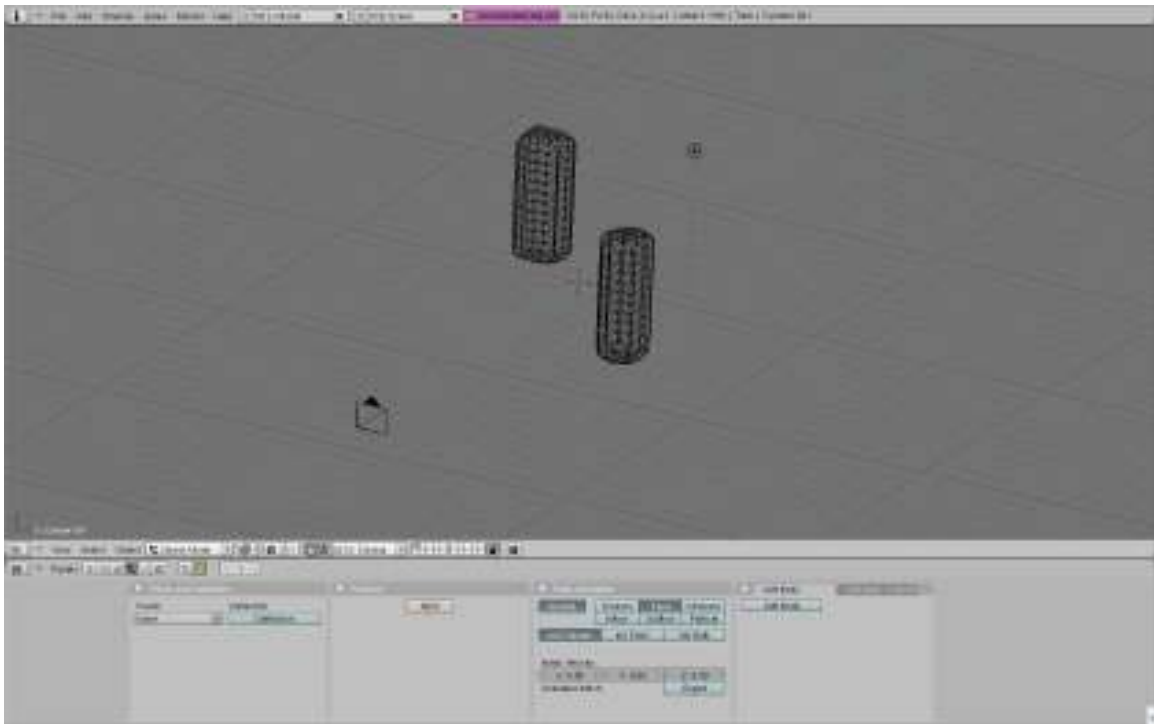


Figure 965 What you should have now

1. Now we are going to rebake the water at a higher resolution for a more detailed result. Select the goo looking mesh, (both blobs should turn purple if either is clicked) and go the fluid simulation tab again. Turn the resolution up to about 100 (or whatever you want) this rebakes the fluid for more detail. If you turn preview mesh up too, you will see it more detailed before rendering. Baking now may take a while.
2. After its done baking, you are ready to change the material of the water to something better looking.
3. First change the renderer to YafRay. Press F10 then under the big Render button select YafRay. Make sure Yafray is installed first, the option still appears if you don't.
4. Change the background white by pressing F5 then click on the box that has world buttons appear when the cursor is hovered over it.
5. You will see HoR, HoG, and HoB. Turn these all the way over to one. The preview should turn white.
6. Now go to the material buttons (red sphere) and make everything look like the following: (I think I circled all the changed options, but maybe not.)



Figure 966 What your material should look like

1. After this is completed look through the frames by pressing right arrow, find one that looks cool then press render and you may get something like this:



Figure 967 Does it look like yours?

Hopefully you got something like that :)

(that's all folks...)

If the render does not look smooth, apply a subsurf.

142.3. Extra Practice

This YouTube tutorial on fluids might also help: [Link¹](#) and this [Realistic Water Texture²](#)

Video tutorial showing fluid simulator objects in Blender 2.49b:
http://www.youtube.com/watch?v=wX5w1Kmsv_A

User Question: When I click Bake there is no box that shows up, I cannot get it to bake my fluids at all, all it does is add a bunch of files to my hard drive and take a long time, What should I do?

¹ <http://au.youtube.com/watch?v=yMBx2pYiqnY>

² <http://forum.nystic.com/viewtopic.php?f=34&t=1445>

Answer: Be sure to set the directory that you would like the files to be saved to, the option to do so is in the area where you changed the box to a domain very bottom there is a folder. Chose a directory and all will be good.

User question: When i click the bake button it says there can't be more than one domain object. I did everything you said to do but i don't understand why it says that.

Answer: The only solution I have found around that is to go to the top header and where it says "SR:2-Model" change that to actor and select the different objects until you find something that doesn't highlight. Delete those. I found that occasionally a domain "disappears" and when you make a new one causes this problem. I have also found it better to set up the domain last around your scene to prevent this annoyance as well.

Note: The domain object does NOT disappear - the Domain object IS the object on your screen which represents the fluid itself.

143. Bump Mapping

This tutorial was created using Blender v2.49

Bump Maps are textures that store the relative height of pixels from the viewpoint of the camera. The pixels seem to be moved in the direction of the facenormals, either in direction to or away from the camera. You may either use greyscale pictures or the intensity values of a RGB-Texture (including images).

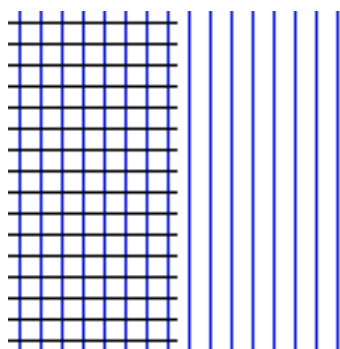


Figure 968 Image 1a:
A texture grid for bump mapping.

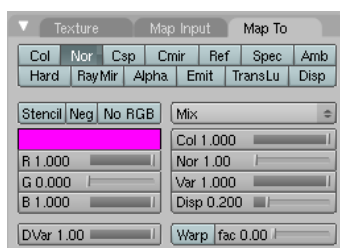


Figure 969 Image 1b:
Material settings for the texture.



Figure 970 Image 1c:
The texture grid applied to a sphere and to a plane.

Bump Maps are easy to apply. They work well on flat surfaces, only to some extent on curved surfaces. On curved surfaces it is more easily noticeable that you don't create real 3D structures. The visible effect depends on factors like lighting, specularly of your material, camera angle, distance and so forth.

- Add a new texture to your object.
- Activate the *Nor* button in the *Map To* panel.
- Set the depth of the bumping with the *Nor* slider.
- Use the texture type *Image* and load your bump map.

Bump maps should contain hard transitions between black and white. A gray wedge (e.g. a linear blend texture) would be hardly visible in the rendering.

143.1. Creating Bump Maps

You can easily create Bump Maps with Blender yourself, this is especially useful if you have modeled some small details on a surface and you realize at the end that your scene will get too complex. You could also use the Bump Maps in a 2D application like Gimp or other, similar programs.

I will create an animated bump map in highest possible quality, it is not always necessary to make such an effort. The goal is to create a wave effect and make an image sequence of it, to be used as a bump map. The original object has 600.000 vertices, the object the map is applied to has 8 vertices.

143.1.1. Setting up the scene

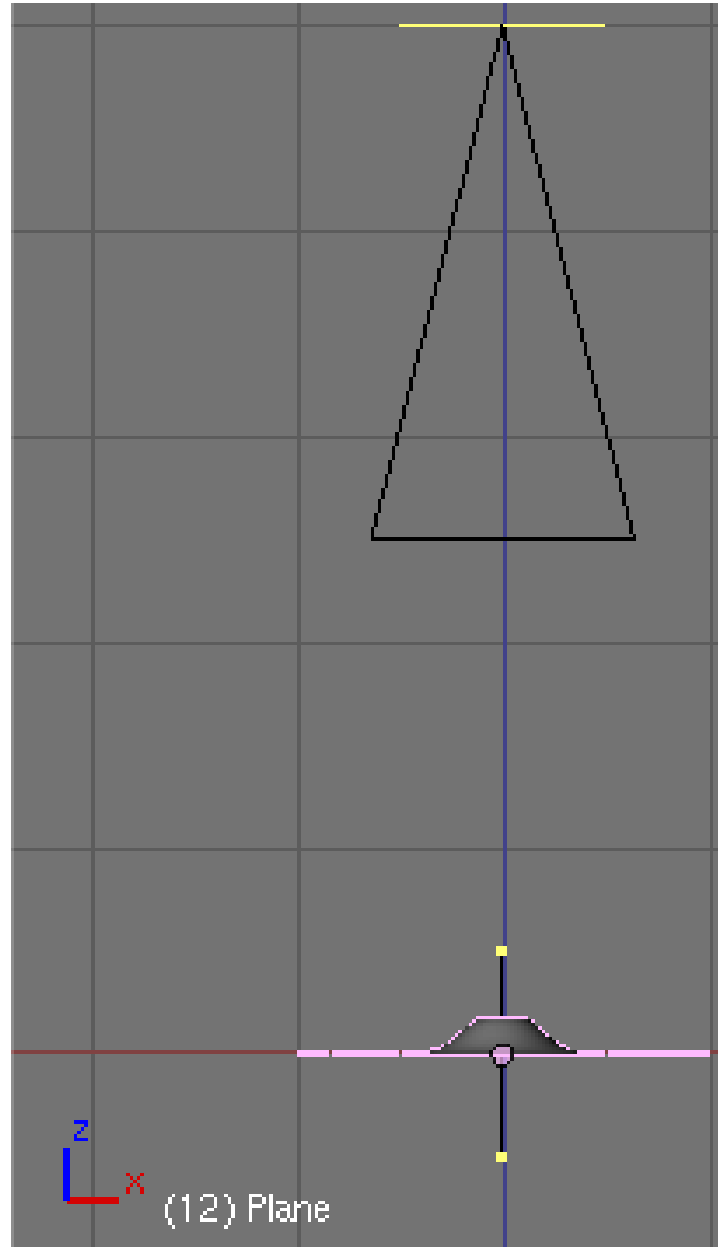


Figure 971 Image 2a: Setting up camera and object in the 3D view.

- Open the default scene and remove the cube.
- Insert a plane and change to edit mode.
- Crease its edges with +1 (**Shift-E**).

- *Subdivide multi* with 11 cuts.
- Change to object mode.
- Add a *SubSurf* modifier with a render level of 6.
- Add a *Wave* modifier.
 - *Time Sta:* -10
 - *Height:* 0.2
 - *Width:* 1.5
 - *Narrow:* 1.5
- Set the camera dead upon the plane.
 - *X/Y/Z:* 0/0/5. You can use the *Transform Properties Panel* to bring the camera to a certain position, the Z-Position is important in this case.
- Set the camera lens to 80.00 (*Editing Buttons->Camera Panel*).

The plane should now fit exactly into the camera view.

- Save your file.

We're going to use the Z-Buffer information to create the bump texture. The Z-Buffer contains the distance from the camera, this is exactly what a bump map is. To render the Z-Buffer information as an image, we're going to use *Composite* nodes. To get the highest possible quality, we will use *Open EXR* as file format, this allows us to store the Z-Buffer information with a numerical accuracy of 32-Bit floating point, instead of a meager 8-Bit value.

143.1.2. Render settings

- Change to the *Anim* tab buttons (f10).
- Set *End* to 40.
- Activate *Do Composite*.
- Select *Open EXR* in the *Format* panel, set *SizeXY* to both 600 (square image for the square plane).
- Set the *Output* directory to *//BumpAnim/*. This creates a subdirectory to the file where the image sequence will be stored. Don't omit the slashes.

143.1.3. Node editor

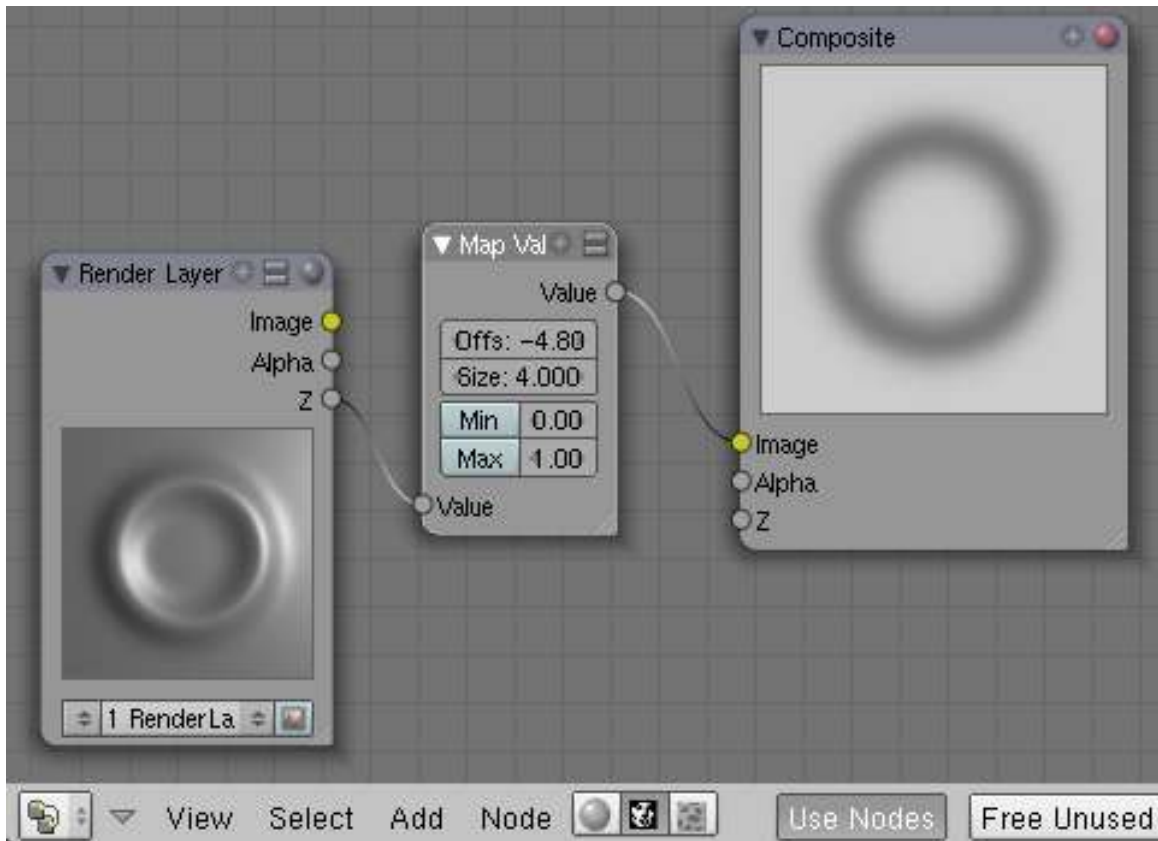


Figure 972 Image 2b: Composite Nodes setup to render the Z-value.

Now the setup for the composite nodes.

- Open a *Node Editor* window.
- Select *Composite Nodes*.
- Activate *Use Nodes*.

A *Render Layer* and a *Composite* node will be inserted automatically.

- Add->*Vector*->*Map Value*.
- Connect the *Z*-Output of the *Render Layer* to the *Value* input, and the *Value* output to the *Image* input of the *Composite* node.

If you render now the image is plain white, we have to talk a bit about the *Map Value* node. The *Offs* value is the distance from the camera where the *Z*-Buffer should start (in negative BU). It is not to important to get the best range, because we use OpenEXR, but if you would like to use PNG instead, you have to select this value carefully.

- Set *Offs* to *-4.8*. This is the smallest distance from the camera needed. You can measure this value if you use the *Camera Clipping*, or if you render and move your mouse with pressed **LMB** above the render window.
- Set *Size* to *4*. The size value is a bit complicated. This is not directly the value range, instead $range = \frac{1}{size}$, and $size = \frac{1}{range}$.

- If you want to include the Z-Values from -4.8 up to -5.05 the range is 0.25, so *Size* is 4.
 - If you want to include the Z-Values from -4.5 to -3.5 the range would be -1, so *Size* would be also -1.
 - If you want to include the Z-Values from -4.5 to -5.5, the range would be 1, so *Size* would be 1.
- Render the animation.

143.2. Applying the animated bump map

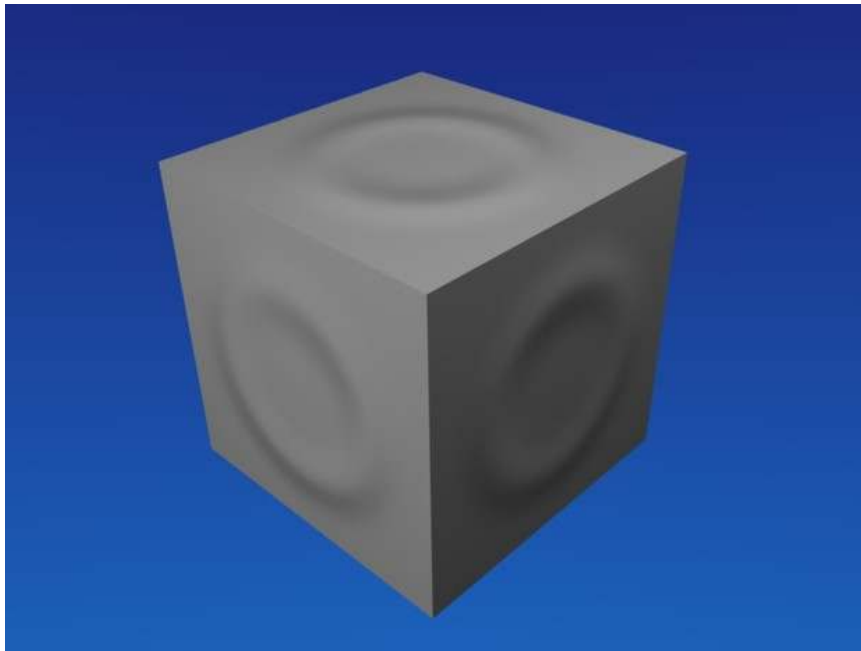


Figure 973 Image 3a: The Bump Map applied Video^a.

^a <http://www.vimeo.com/4959751>

- Open a new file.
- Select the default cube and change to the *Texture* buttons.
- Use the texture type *Image*.
- Load the first OpenEXR-Image from the directory where you have stored the sequence.
- Activate *Sequence* in the *Image* panel, set *Frames* to 40.
- Change to the *Material* buttons.
- Activate *Nor* for the texture and set the *Nor* slider to 25 in the *Map To* panel.
- Set the mapping to *Cube* in the *Map Input* panel.

This is it. Pretty much the same look as with the real deformation, but using much less resources.

144. Normal Mapping

Noob Warning: This technique is best used for models and animation, and is not well used and applied in the Blender Game Engine

Normal Mapping in 3D graphics development is the process of using an RGB¹ color-map to create a three-dimensional relief on a two-dimensional plane. The source for the normal map in blender can either be a texture already installed in Blender, or using an external picture-file (.jpg, .jpeg, .bmp, and so on) and loading it as a normal map. One great aspect of Blender and normal maps is that Blender can very easily be used to **create** normal maps that can be turn around and used in blender to reduce poly count. This tutorial will show the various ways to create a normal map and how to apply different normal maps to your model in Blender.

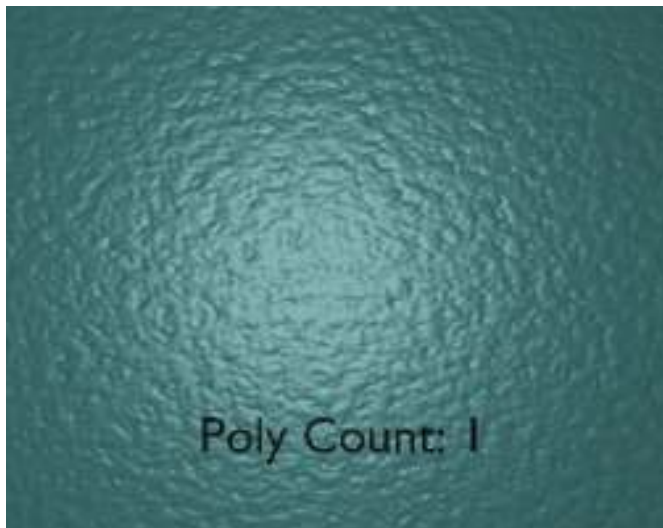


Figure 974

¹ <http://en.wikipedia.org/wiki/RGB>

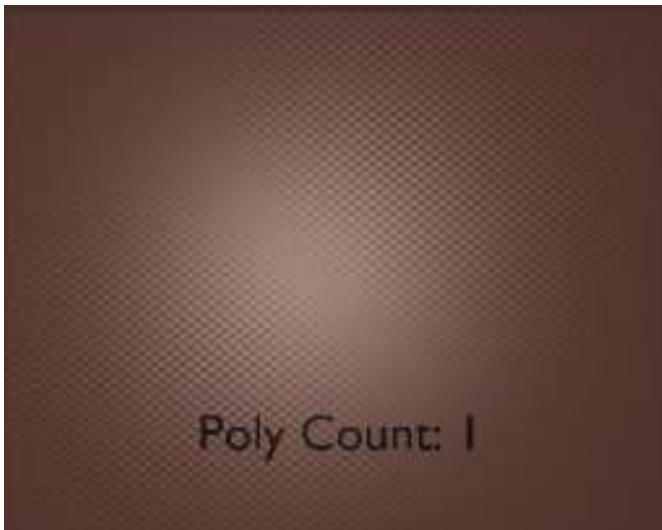


Figure 975

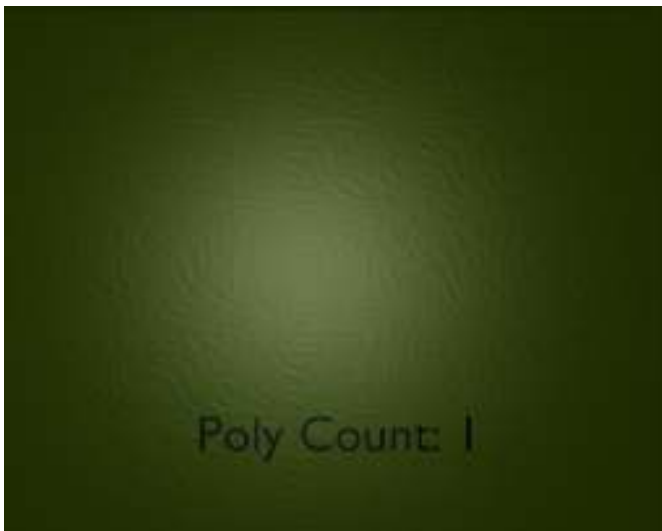


Figure 976

Sections within Chapter:

1. Texture Normal Mapping²
2. Color Map Normal Mapping³

2 Chapter 145 on page 1133

3 Chapter 146 on page 1137

145. Texture Normal Mapping

145.1. The Texture-Normal

To familiarize yourself with normal maps and how they work, we will begin by using the texture engine in Blender to create our first normal map. This process is very useful for creating quick and non-specific normal maps for your projects that need a bit of texture.

To begin, open blender and whatever your basic load settings are, in the 3D view, delete everything so the field is clear. (**AKEY** to select all, then **XKEY** to delete)

Move the 3D cursor to the center of the X,Y, and Z axis, and switch to top view (**NUM7**) orthographic view (**NUM5**). Hit **Shift-SKEY-4KEY** then **CKEY** to center your view now.

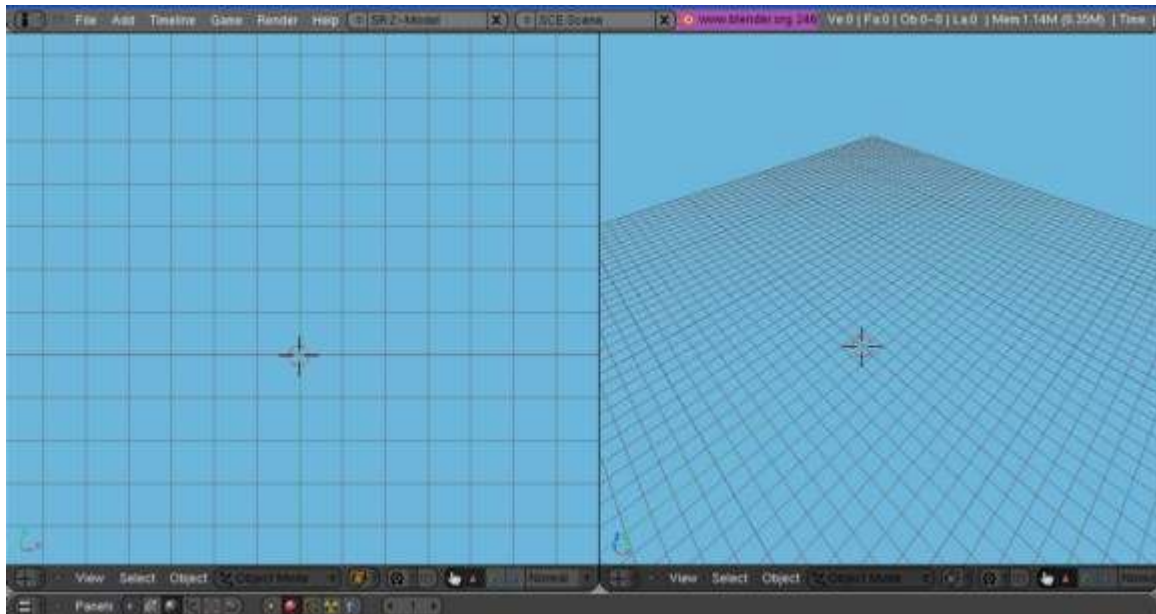


Figure 977

Add a mesh plane to the field, and scale that plane to five times normal size. (**Shift-AKEY**, select *Add->Mesh->Plane*, **TAB** out of edit mode, **SKEY** then **5KEY** to scale five times size.) This plane is going to be our base-it will remain untextured so it can be used as a comparison. Don't give it a material.

Now, with the 3D cursor still on the center, add a mesh cube to the field. Scale this cube 3 times in the X and Y directions, and 0.5 times in the Z direction. (**Shift-AKEY**, select *Add->Mesh->Cube*, **TAB** out of edit mode, **SKEY**, **Shift-XKEY**, then **3KEY** to scale 3 times in X and Y directions, **SKEY**, **ZKEY**, then **.KEY-5KEY** for Z.) This cube will serve as our texture comparison.

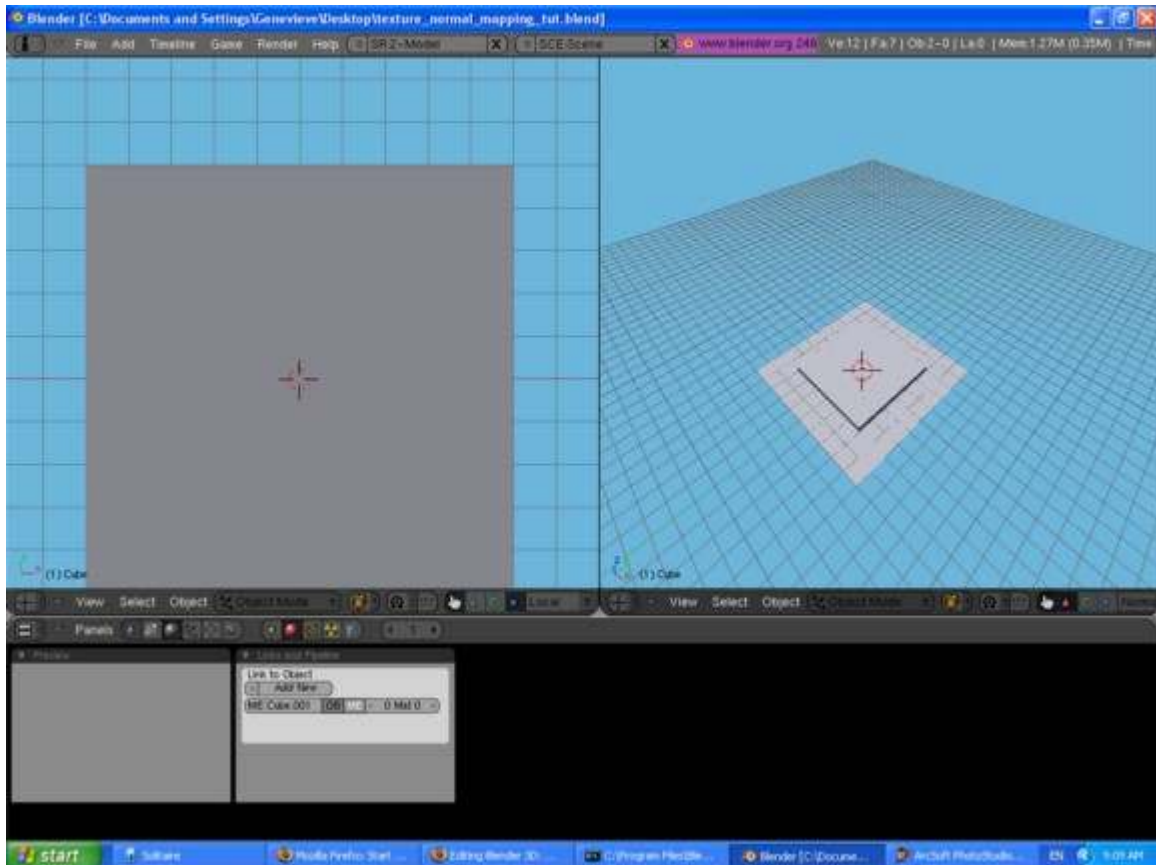


Figure 978

For this tutorial, let's use the "Musgrave" texture to make a 3D relief.

Select the cube, and then in the Buttons Window select the material index. Give the Cube a material, and make this material any light color you wish to differentiate it from the base plane (darker colors won't show the relief that well). Now select the texture icon in the material window. Add a new texture, and make that texture Musgrave.

Make the settings for Musgrave the following:

- Type: Multifractal
- H: 0.5
- Lacu: 3
- Octs: 2
- iScale: 0.45
- NoiseSize: 0.075
- NoiseBasis: Voronoi F2
- Nabla: 0.025

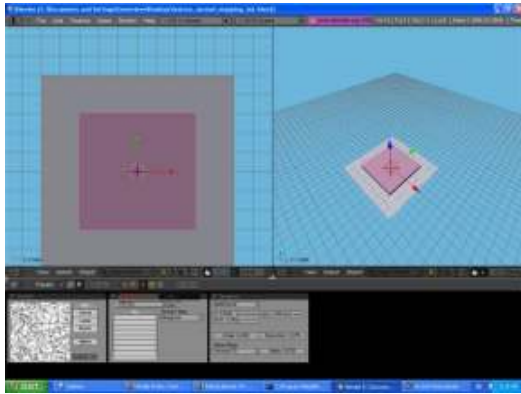


Figure 979

Excellent, now we are ready to show what we have done! If you return to the material window in the material index, then you will probably be wondering what's wrong-your preview window just shows a sphere with a huge purple splotch! Don't worry, we just haven't applied the texture as a normal map yet, it's still just a color map. Click the "Map To" tab in the material window and you will see a whole row of buttons, menus, and sliders. Currently on the top row of buttons the "Col" button is depressed, which is how the texture knows to apply as a color map. Click the "Col" button to remove the color from your preview window. Now click and depress the "Nor" button and look what happens! An intricate texture embeds itself on the sphere in the preview window. The Musgrave texture has caused a depression texture within the preview. If you happen to click the "Nor" button again, you will see the letters turn yellow and now that means the Musgrave texture is causing an elevation texture. Click it once more and the texture is turned off. Keep the "Nor" button depressed for this tutorial.



Figure 980

Now that you know how that works, let's see what it did to your cube. First, go to front view (numpad-1) and zoom out a bit so about 10% of the screen is filled with your cube. Select the cube and hit Shift-S-key-4-key then C-key to center your view. In the upper-left corner, add a sun-lamp (Shift-A-key, Add->Lamp->Lamp). Adjust the "Dist" Slider to a very far length, to make sure it reaches the cube. In the upper-right corner (make it a little closer to the cube) add a Camera (Shift-A-key, Add->Camera).

Move the Camera to get a better perspective of your work, by rotating it to look at the cube and the plane. If you want exact directions: Switch to top view (numpad-7). Move -4 Units in Y direction. Rotate -75 degrees on Z rotation. Rotate -45 degrees on Y rotation. Now hit numpad-0 to see how well pointed it is to the cube. Make adjustments to your own work if you'd like.

Now you are ready to Render and see how it turned out!

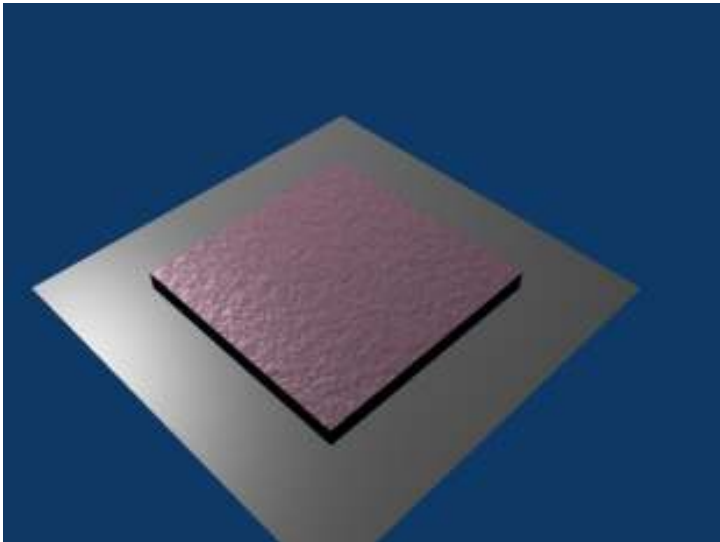


Figure 981

After your first render, I'll let you know that you can adjust the depth of the shading. Select the cube, and in the material index in the "Map To" tab, in the mid-right area, there is a slider called "Nor" that should be set at 0.5. Just to test, set it to 0 and render to see what happens. Then set it to 2 and render to see the results.

After you see it rendered, you probably realized that the edges of the cube were very sharp and didn't have the rocky texture. This is because the 3D relief on the cube is fake--it's not actual 3D, it's the computer just calculating where light should bounce off and in what direction. Imagine how many faces it would have taken to create this texture shading without Normal Mapping.

If you want, you can move the lamp around and see how the light moves around the fake-bumps.

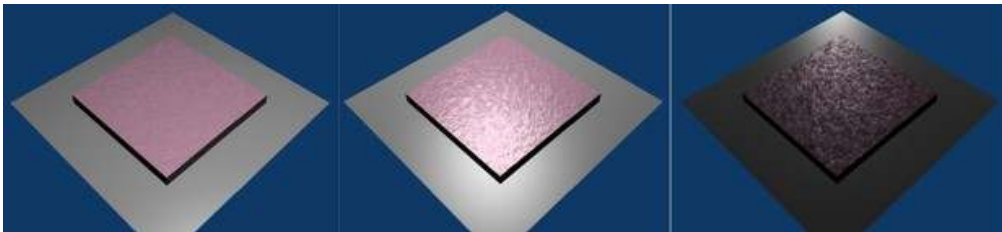


Figure 982

This is best if you just need a non-specific texture, such as asphalt, cement, carpet, or even fabric. Now if you want to do something like make a low-poly brick wall, you will need to a specifically-tailored normal map for such thing. The next section will show you how you can make your own normal map and apply it as well.

146. Color Map Normal Mapping

146.1. The Color-Map-Normal

For more advanced Blender users, you probably have at one time or another wanted to create something very detailed and realistic, only to realize that one modeled head or one cool building has taken an enormous amount of memory, and rendering (or worse, animating!) something like that would probably take days! Well there is an awesome trick you can use to create intricate details and design shading, with a relatively low amount of faces. You can use blender to create and apply normal maps, which are RGB colored maps that can be used as a texture to calculate how light will bounce off an objects surface. All you need is blender and photo-editing software to tweak the picture a bit.

146.2. Creating a Normal Map

You can very easily create a normal map once you figure out all the necessary settings. You can start by either making a high-poly count version of your object, or making the material that will be applied to the object to make the normal map.

146.2.1. High-Poly Version

We will first start by making a simple object with a high-poly count. Start up blender, and let's delete everything there first, then go to top-view, orthographic, and start by adding a camera.

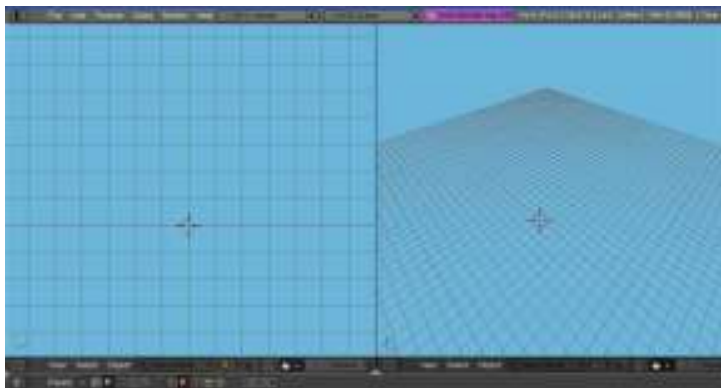


Figure 983

Next, start with a circle, with a radius of 2 and 32 vertices in circumference. Keep it filled. Select the edges of the circle, keeping the inner vertex unselected, and extrude edges only. Scale the new

vertices by 1.1. Repeat this with only the outermost vertices 6 more times, until you have a ring of 7 vertices and a center vertex. Now, we will refer to the center vertex as center, the innermost ring as Ring 1, and the outermost ring as Ring 7. Deselect all vertices, then select all vertices of ring 2, 4, and 6 (This can easily be done by holding the Alt and Shift keys, then selecting an edge adjoining two vertices of the same ring for each ring). With these rings selected, move it 0.4 in the Z direction. Now, let's smooth this bowl out. In the editing index, under the modifiers tab, add a subsurf modifier of 2, and then apply the modifier. Then under the links and materials tab, select Set Smooth. Next after the object looks all nice and smooth, select your Camera and move it positive in the Z direction so it's positioned directly above the object looking straight down at it, and edit the camera Lens properties so that it is set to Orthographic, and scale the lens so that it includes all the bowl and fills up the camera space as much as possible.



Figure 984

146.2.2. Creating the Color-Map Material

Now, to be able for the Render to apply the colors on the right areas of the object. For this, we will start by applying a material the bowl. Make this material 0.0 on the R, G, and B spectra, and depress the shadeless button. Now, go to the textures index and add 3 textures to the material, and set them as "Blend" textures with "Lin" depressed (a linear calculation). For these 3 textures, name them something to do with either "Red"/"Green"/"Blue" or "X"/"Y"/"Z" in that order, because doing this will ensure you keep the channels in order. The following image will show all settings that need to be set to make the material work.

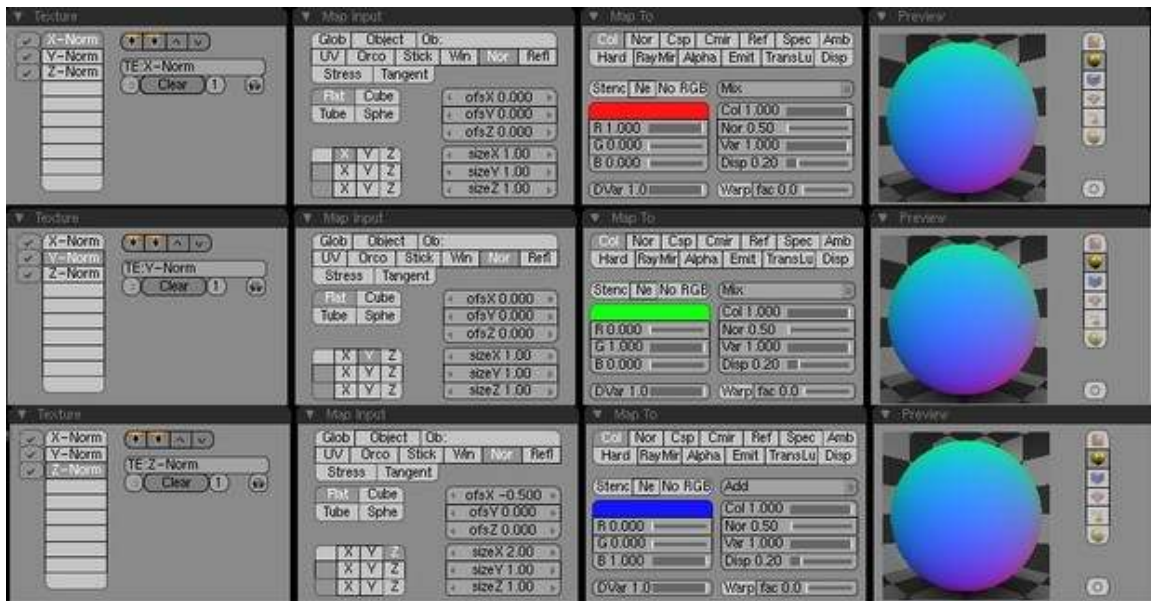


Figure 985

Now before your Rendering you want to make the World texture to be completely black, so this way it won't interfere when you apply it as a normal map. Now Render your image, and hit F3 to save. It should come out to something like this.

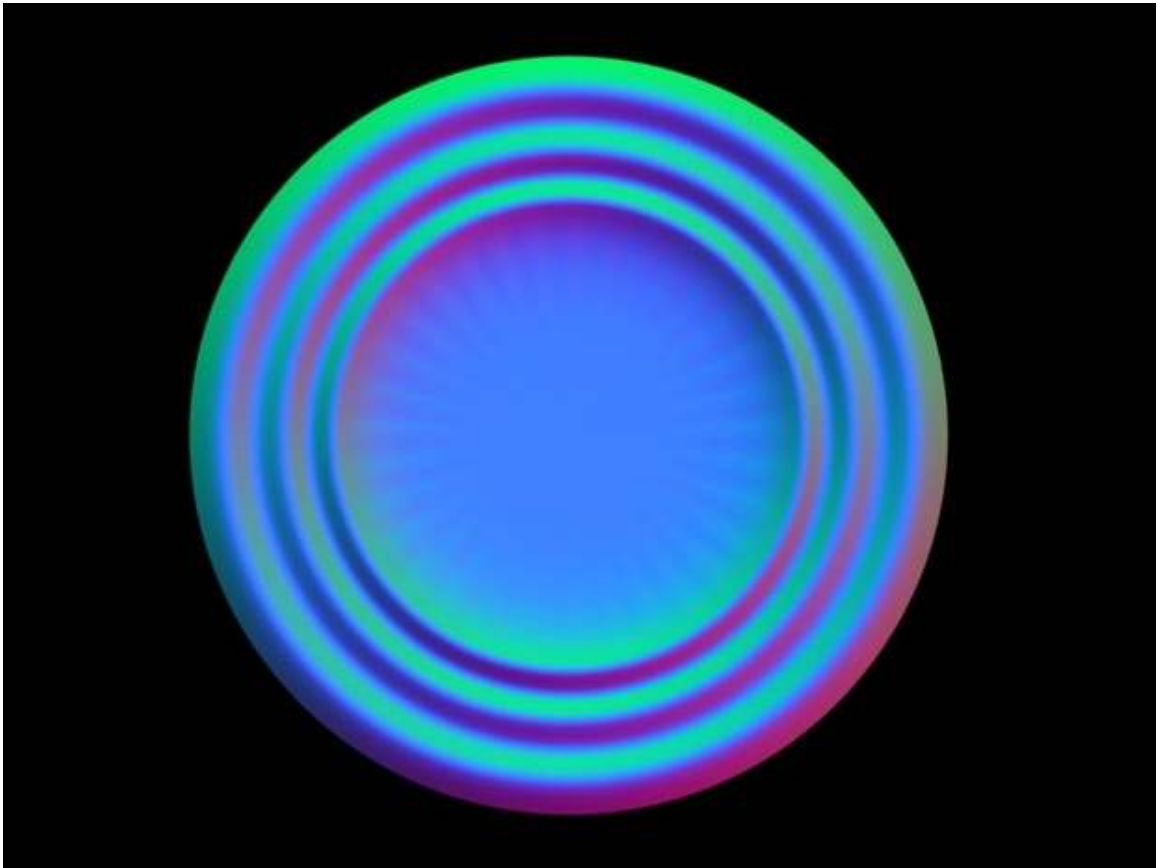


Figure 986

As a final note in this subsection, what I suggest doing is deleting the bowl and saving the file as something like "BlenderForNormals.blend" so if you ever want to make another normal, the area and materials are already set up and all you need to do is import or create the object. Just a suggestion.

146.2.3. Applying as a Normal Map

Well we got the image, now let's apply it and make it come out as a low poly-count texture. So, let's continue by deleting the high poly bowl and instead add a plane in the middle of the camera. Now, apply a material to the object and make the material hard, soft, rocky, wet, or however you want it to look. After you've made the plane how you want, add a texture to the material, and make it an image texture. Load the image as the texture, depress the normal map button, and set the normal spacing to object. Now, on the materials index under the Map To tab, make sure the Col button is relieved and instead depress the Nor button. Just add a lamp above the plane, and Render! You should now have the same shading effect as the original bowl, with about 0.01% of the original faces--one.

If your Render comes out odd, it may be because the shading is being applied too heavily. You can alter this by adjusting the Nor slider directly below the "Mix" drop-down menu. O used about 0.2 and it came out just fine, like this picture.

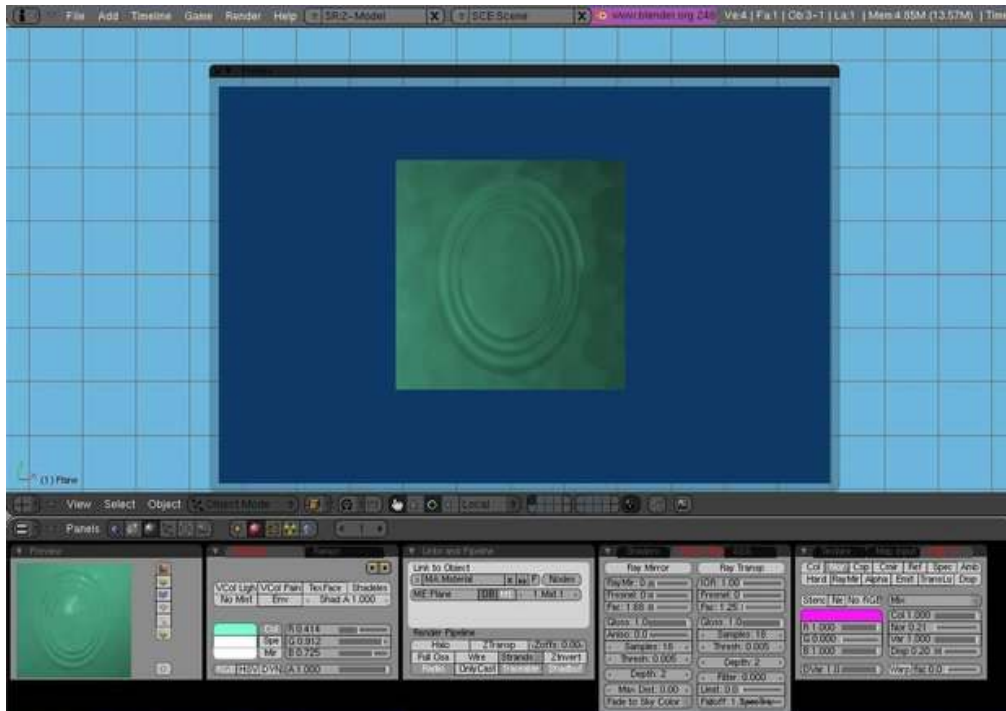


Figure 987

Congratulations, that's all there is to normal mapping. It's a simple technique with many great uses!

147. Nodes

Blender's *Node Editor* lets you assemble various processing blocks (*nodes*) into combinations which feed data to one another along connections that you specify to produce complex effects. These effects can be used in three different ways: as textures, as materials, or for compositing. The Node Editor makes different kinds of processing blocks available, depending on which of the three kinds of effects you are producing.

148. Texture Nodes

Texture nodes allow you to produce textures that are the result of complex computations. This tutorial will just scratch the surface of what's possible.

148.1. A Simple “Rainbow” Texture

Start with a new Blender document. Delete the default cube. Add a new plane object. In the Material buttons, click Add New to assign a new material to the plane (the default name of this should be “MA: Material”). In the Texture mini-window, click Add New to assign a new texture to the material (the default name of this should be “TE: Tex”). Now split the area showing the 3D view into two side-by-side areas. In the right-hand area, bring up the Node Editor. In its area header, the group of 3 little icons next to the menu names has tooltips “Material Nodes”, “Composite Nodes” and “Texture Nodes” respectively; click on the third one for Texture Nodes. You will see appear, further along the area header, a popup menu containing the names of the textures so far created in your document: this should be showing the name “TE: Tex” of the texture you assigned to the material for your plane object. And to the right of that, there should be a button titled “Use Nodes”. Click that, and you should see a pair of initial nodes immediately appear in the editor: an *input* node titled “Checker” (the name may be truncated to “Che” in the initial sizing of the node window) and an *output* node titled “Output”, with an editable field in it containing “Name: Default”.

There must be at least one output node in the texture definition; the data fed to this will make up the final texture. The names you assign in the Name field will appear in the Texture mini-window, in a popup menu titled “Use Output”, with one item for each output node. Initially this will contain just one entry, the one named “Default” by default. We'll leave it that way.

Each node window has little coloured circles (terminals) on its left and right edges; the ones on the left edge (if any) are inputs for feeding data from other nodes, and the ones on the right edge (if any) are for supplying data to other nodes. Thus, output nodes have no outputs (they're the final destination for the data), while input nodes have no inputs. Other node types represent intermediate stages in the processing, so they will have both inputs and outputs.

Back in the Node Editor, click on the titlebar of the “Checker” node window and use either **XKEY** or **DEL** to delete it. Note that, unlike deleting objects in the 3D view, there is **no** confirmation popup: the node immediately disappears. (Of course, you can use **CTRL-ZKEY** to undo operations in the usual way.)

Now go to the “Add” menu, find the “Input” submenu, and select the “Coordinates” item. This will add a new input node, which just produces the unadorned texture coordinates as its data. You will see it has one output terminal, titled “Coordinates”. Left-click on this, and drag to the input terminal in the Output window with the word “Color” and a small rectangular colour swatch next to it. Voilà! A line (effectively a wire) should appear connecting the two terminals, and you should see the big

square colour swatch in the Output window change from black to a whole rainbow of colours. That's your texture!

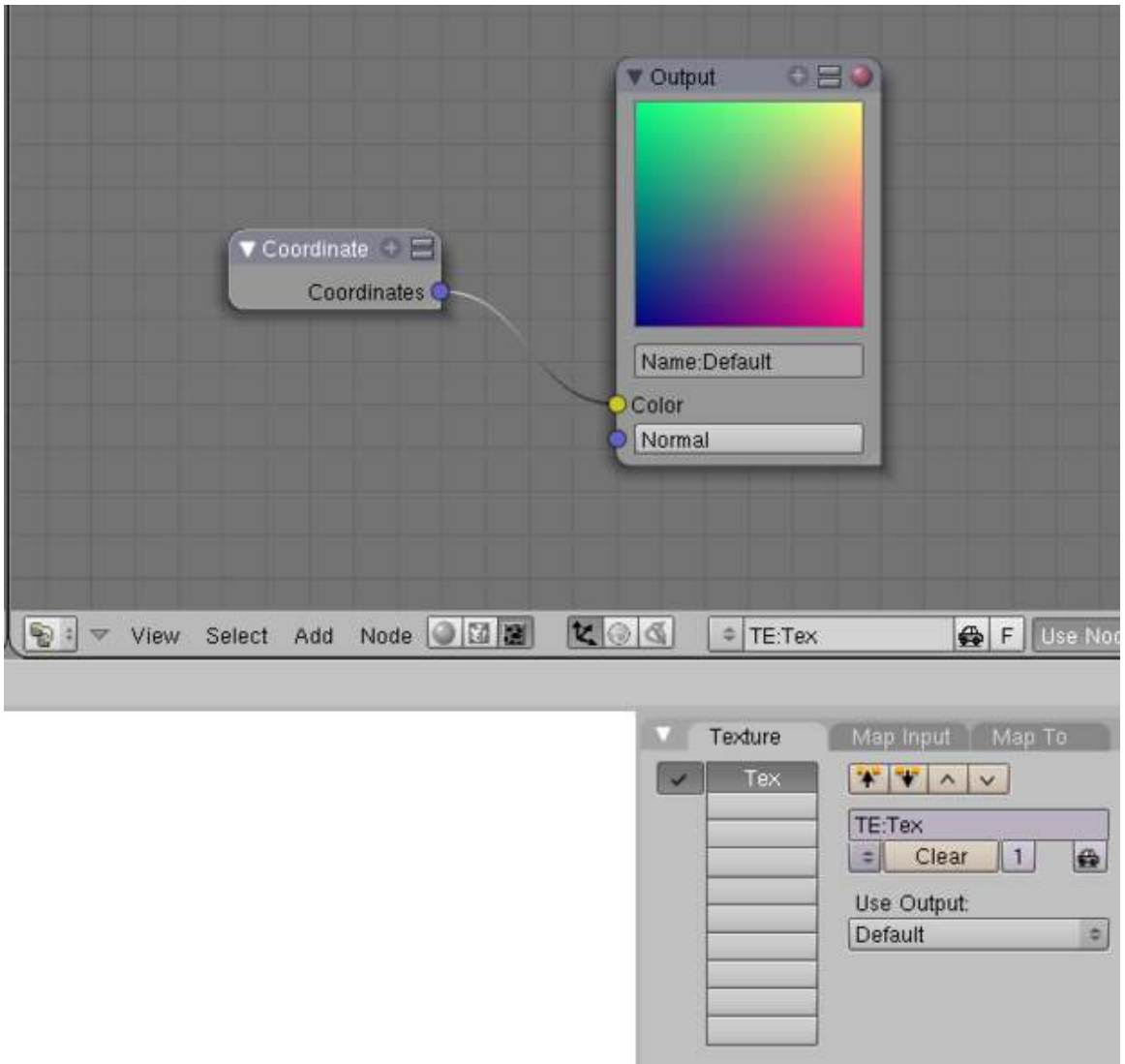


Figure 988

If you make a wrong connection, you can cut it by dragging across the wire with **LMB**; when you let go the button, the wire will disappear.

At this point, you should be able to hit **F12** to render, and you will see your plane object with the rainbow texture applied. You can see why I chose a Plane object: being flat, the entire texture is visible from one camera angle.

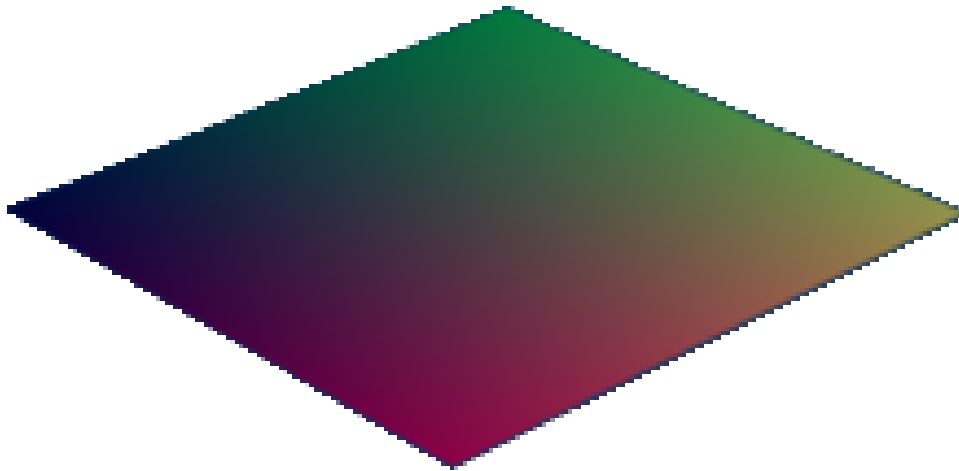


Figure 989

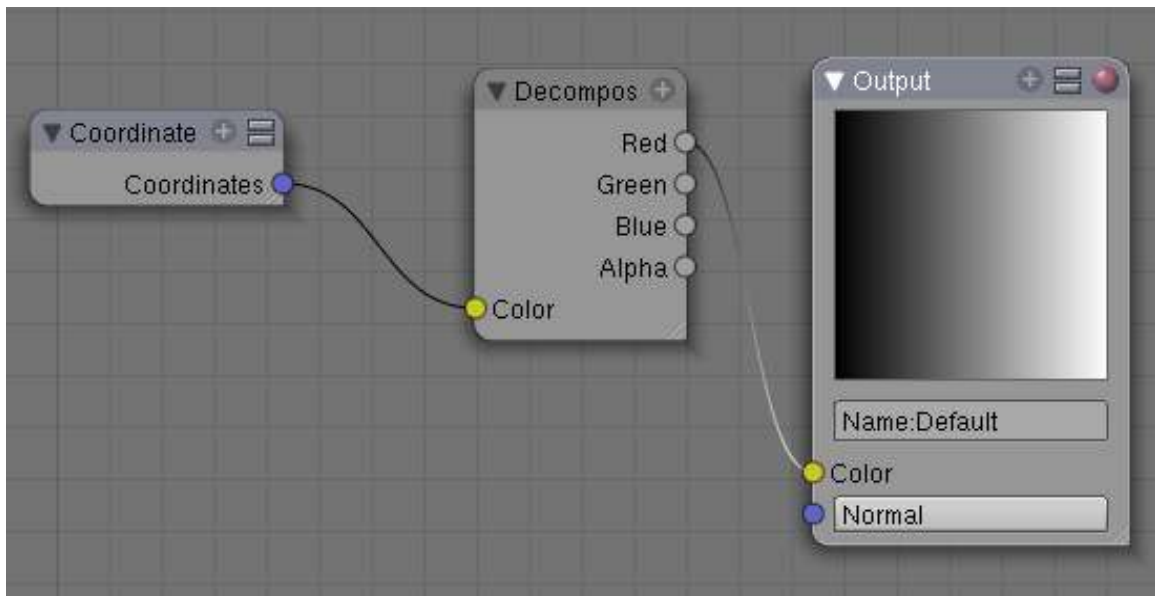
Alternatively, in the 3D editor, you can change the Draw Type to Shaded, and you should be able to see the texture immediately applied there. However, there seems to be a bug in version 2.49b of Blender where changes to the texture definition are not automatically visible in the 3D view, until you make some slight change to the view, like for example rotating it slightly. (After applying the random view rotation, you can hit **NUM7** to restore the default top-aligned view.)

148.2. Scalars Versus Vectors

The wires carry numbers between nodes. These numbers can be of two kinds: a *scalar* is a single real quantity, like “0.5”, while a *vector* can consist of two, three or four scalars in a sequence, like “(0.5, 0.75, 0.2, 1.0)”. The number of components in a vector is also known as the *dimension* of the vector. Two-dimensional vectors can be used to represent texture coordinates, while three-dimensional vectors can represent positions in space. Colours can be represented with 3 dimensions (R, G,B or H, S, V) or 4 dimensions (RGB or HSV plus alpha channel).

Some nodes operate on scalars, while others operate on vectors. And there are nodes where some terminals input or output scalars, while others input or output vectors.

Try this. To the above Coordinates and Output nodes, add a “Decompose RGBA” node (under the “Color” submenu of the “Add” menu). Feed the output from the Coordinates node to the “Color” input node of the Decompose node. Now connect just the “Red” output terminal from the Decompose node to the output node:

**Figure 990**

See how the Output swatch shows just a greyscale ramp running from black on the left to white on the right? This shows how the first dimension of a vector is interpreted as the red component in an RGB colour, but as the X coordinate in a position. Remove the output from the Red terminal, and take it from the Green terminal instead; now you will see the ramp going from black at the bottom to white at the top. The second dimension of a vector is the green component in an RGB colour, and the Y coordinate in a position.

As far as Blender's Node Editor is concerned, colours and positions are not different types. Both are vectors, and one can be interpreted as the other. This can be (ab)used for some creative effects! For example, the name of the "Decompose RGBA" node says it's for decomposing an RGBA colour into its components, but here we are using it to take apart a coordinate vector instead.

Note also in the above, that the Color terminal on the Output node expects a vector RGB colour, but when you feed it a single colour component (i.e. a scalar), it simply replicates it the necessary number of times to make a vector of the required dimension. Since all components of the RGB colour are the same, you get a shade of grey.

149. Material Nodes

Material nodes look very similar to texture nodes, and indeed there is a fair amount of overlap in functionality. Some important differences are:

- Functions for performing spatial transformations (e.g. rotation and scaling) are only available for texture nodes.
- Functions for altering diffuse versus specular colours and rendering parameters, like translucency, are only available for material nodes.

149.1. A Simple Graduated Material

First of all, let's set up a really basic modelling scene to show off the node-based material you'll be creating in a moment.

Open a new Blender document, with the default cube, default light and default camera. Create a plane object. Rotate and position that so it lies behind the cube from the camera's viewpoint. Scale the plane up to say, 4x, to form a decent-sized backdrop. Assign it a new default material, and a texture with some detail to it; I used a simple Marble with default settings, except under the "Map To" miniwindow I changed the colour to a medium grey from the default magenta, to stop it hurting my eyes.

Now select the cube. Scale it 3x in the Z-axis; the elongated proportions should show off a gradation more nicely. In the "Material" miniwindow, set the "Col" (diffuse) colour to white, and the alpha to zero. In the "Mirror Trans" miniwindow, turn on "Ray Transp" and set the "IOR" to some suitable refractive index, say 1.5 for glass.

Hit **F12** to do an initial render, just to confirm you've got everything nicely arranged.

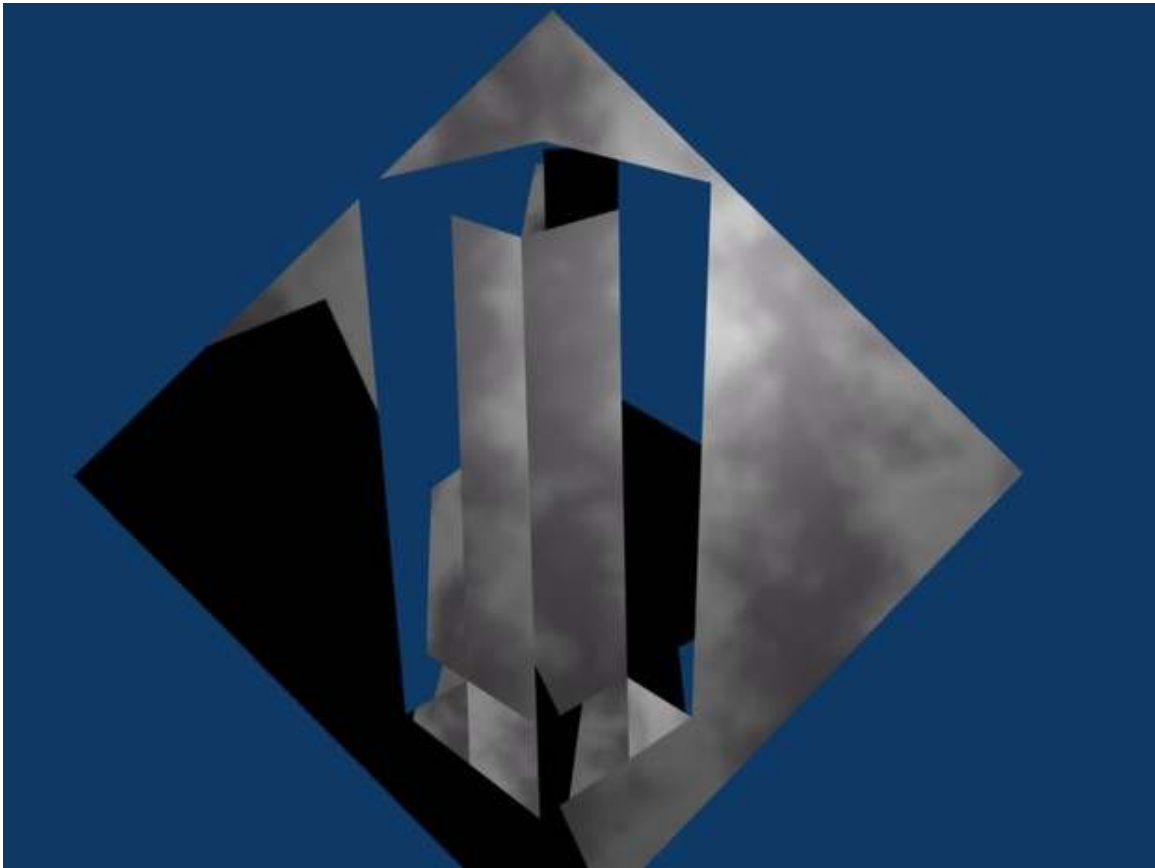


Figure 991

With the cube still selected, open the Node Editor. By default, the Material Nodes mode should already be selected, and the combo box should be showing the name of the material you assigned to the cube; select it if it's not. Click the "Use Nodes" button, and immediately two node windows should appear, a Material one and Output one.

Click on the title of the Material node window, and use **XKEY** or **DEL** to get rid of it. Go to the "Add" menu, "Input" submenu, and add a new "Geometry" node. Also add a new "Extended Material" node. And from the "Convertor" submenu, a new "Separate RGB" node. In the middle of the Extended Material node window, there is a popup menu showing you the names of the materials in your document; select name of the material you previously assigned to the cube again.

Make sure there are no wires running between any of the nodes; now run one from the "Color" output terminal on the Material node to the "Color" input on the Output node, and one from the "Alpha" output on the former to the "Alpha" input on the latter.

Notice that, next to each of the input terminals in the Material window, there is either a field for entering a scalar value (grey terminals), or a button which pops up a little set of controls for letting you specify X, Y and Z components for a vector (blue terminals), or a colour swatch which pops up to let you pick a new colour (yellow terminals). This way, you don't have to make connections to all the inputs, or indeed any of them; any unconnected inputs will be set to the constant value that you specify.

Anyway, connect a wire from the “Global” output of the Geometry node to the “Image” input of the Separate RGB node. “But geometry isn’t a colour!?” I hear you cry. But as explained on [Texture Nodes¹](#), a vector is a vector. Feeding in a position instead of a colour to Separate RGB means its outputs are not the R, G and B components of the colour, but the X, Y and Z components of the position. So run another wire from the B (i.e. Z) output of Separate RGB to the Alpha input of the Extended Material window. The resulting layout should look like this:

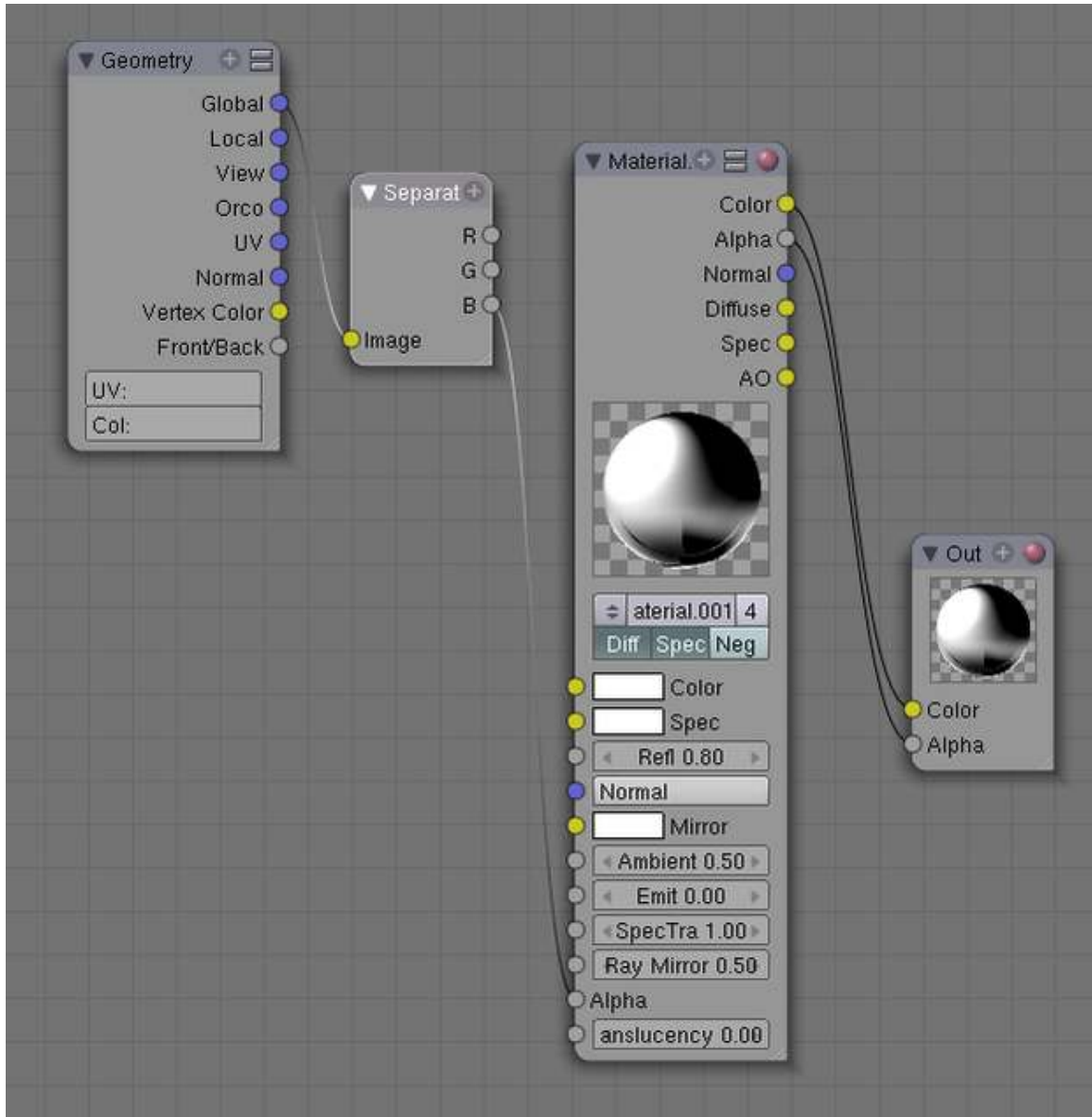


Figure 992

Now you’re ready to render. But before hitting **F12**, hit **JKEY** to switch render buffers. That way you still have the previous render to compare with. Once the render window comes up, you can hit **JKEY** to alternate between the two renders.

¹ Chapter 148 on page 1145

Here's what my node example looked like (note I turned the energy of the lamp down to 0.5, otherwise it seemed too bright):

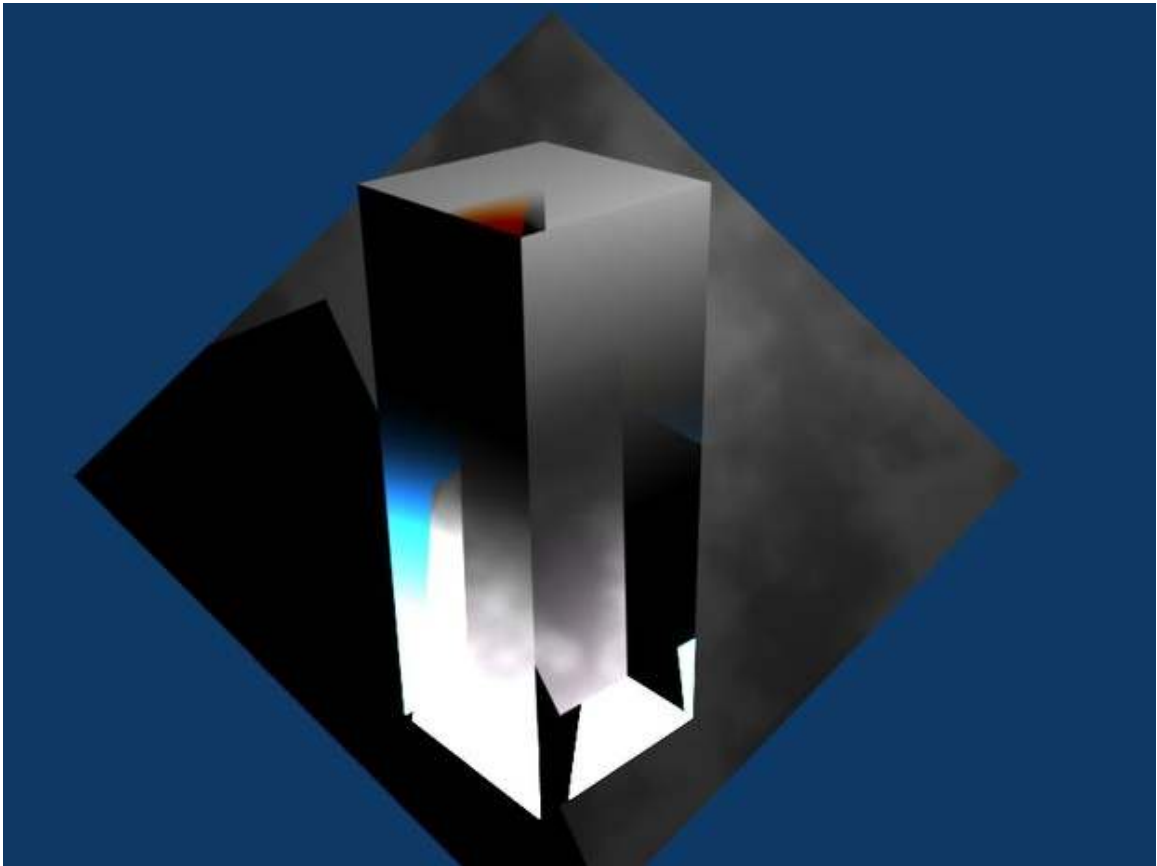


Figure 993

Notice the gradation of transparency along the Z-axis, from fully transparent at the bottom to fully opaque at the top.

For added fun, try taking the material Alpha setting from the R or G outputs of the Separate RGB node instead, just to see the gradation orient along the X or Y axes.

150. Compositing

Compositing is the combination of multiple sources of visual input into a single, final image. This is common functionality in expensive, commercial video editors. Some of these (such as Adobe After Effects) use a layer-based compositing engine. Others (such as Apple Shake, The Foundry's Nuke, and eyeon fusion) use a more powerful node-based compositing engine. Blender features a very advanced and powerful compositing engine through the use of the Nodes Editor window. Although the challenge of learning to use nodes may seem intimidating, it can be very rewarding because of the professional-looking final products that can be easily produced.

150.1. Introduction to Composite Nodes

There are several types of nodes, and each has a different purpose. When nodes are first enabled (described later), two types are present: an input and an output. In addition to these, there are nodes that go between these two groups. Images enter a node through the left side (unless the node is an input) and leave the node through the right side (unless the node is an output). Threads of nodes connected together are known as "noodles."

150.1.1. Enabling Composition with Nodes

In the Scene tab (F10), select "Do Composite" from the Animate panel. Switch from 3D view to a Nodes Editor window. Select Composite Nodes (the face button) and Use Nodes. Two nodes should appear; one is a Render Layer node and the other is a Composite node. If you haven't already rendered the scene, do so by pressing the picture in the bottom-right corner of the Render Layer node. The scene will render and appear in both nodes.

150.1.2. Videos

Composite Nodes Video Tutorial (2.49b): <http://www.youtube.com/watch?v=AAAnxoJsdAAM>¹

Vector Blur Tutorial (2.49b): <http://www.youtube.com/watch?v=qY4WcNgEXv8>

Depth of Field (DOF) Using Composite Nodes (2.49b): <http://www.youtube.com/watch?v=HBomEv-bEtw>

¹ <http://www.youtube.com/watch?v=AAAnxoJsdAAM>

151. Introduction

Welcome to the wonderful yet complex world of computer animation! Through these pages I will try to show you everything old and new about the new animation system in Blender 2.4. But, before we get started, there are some basic notions about datablocks you should know. Animation in Blender is based on the fact that you have something moving in a Blender scene. For example, a ball bouncing on a floor plane:

-So you have a scene datablock, which holds some info about the scene itself, as you can see in the Render button window (F10KEY). -You populate this scene with various objects (which in this case refers to containers for data, not the actual mesh data that shapes the object itself). The only goal of an object is to hold the whereabouts of the data you want to see in your scene. It also holds the object *instance's* properties such as "does it have soft body or particle options, and do we draw its name?". Most of the info on an object can be seen in the Object Window (F7KEY).

An object links to all of the data you can see in a 3D view such as **mesh**, **curves**, **nurbs**, **lattices**, **armatures**, **metadata**, the **empty** property, text, camera and lamps.

So the ball you just added to the scene is in fact a mesh, linked to an object that is in turn linked to the current scene.

Now there are also data blocks you can't see in 3D view, such as **material**, **texture**, **Ipo**, **action** and **image**. Instead, you have a special window in which to edit them. This is the idea behind the Blender interface, wherein each data block has a window for you to edit the data.

So back to this bouncing ball: It's also moving across the plane! So an ""Ipo"" data block is linked to the object, telling it where in space the object will be at each frame of the animation. This Ipo is editable in the Ipo window when selecting the ball in 3D view. In Blender, the work you are performing is always on the currently active (selected) object and data.

Looking at the OOPS (object oriented programming system) view (or SHIFT-F9KEY), we can get a good idea of the internal data structure:

Figure 994

Again, you are working in the scene "Scene", with an object "Sphere" linked to the mesh data block "Sphere" and the Ipo datablock "ObIpo". Why is that important? Because from there, you can start playing with the datablocks, linking them all around your projects to reuse old work. For example you can create more than one Ipo, and use the one you want, or tell more than one object to use the same Ipo, or to use the same object in more than one Scene.

Most of the linking job can be done in the Edit button window (F9KEY). Where you can tell an object to use another mesh's data block for Ipo, material, texture or image. There is always a little dropdown menu button for you to select an already-existing data block.

Now, when it comes to animation, you have to understand the way Blender handles data very well, because using Blender is always a matter of plugging data blocks together when working with Ipos, actions and NLA objects.

152. Armature Object

The Armature Object in itself is a tool for the animator to move an object or group of vertices in a reliable way. An armature is made of bones, which can be parented to each other, or connected to each other. It was built with the idea of a skeleton in mind. And in fact, in the 3D world with animation inclusive, its considered to be the skeleton.

You can add it using the SPACEKEY in 3Dview and selecting Armature. You'll then enter into Edit Mode where you can add or move bones to build your default rig. An armature has 3 states. You can switch using the dropdown menu in the header of the 3Dview or use the TABKEY to switch between Editmode <-> [Objectmode|Posemode] and CTRL-TABKEY to switch between Objectmode <-> Posemode:

- Object Mode¹: Your armature is like any other Object, you can move it around the scene, scale it, rotate it and edit options in the button window.
- Edit Mode²: Your armature is in what we call rest position, you can modify the bones it contains.
- Pose Mode³: Your armature is ready to be animated, each bone can be moved, scaled or rotated, constraints get applied, you can pose your character and animate the bones' behavior over time. Take note that Pose mode is now a state of the armature you can switch on/off using CTRL-TABKEY. So when in Pose, you are still in object mode (you can select another object, contrary to the edit-mode)

Note: The following 3 pages of this tutorial contain screenshots and discuss techniques that are only available in Blender 2.40a and later. Refer to the Blender 2.40a release notes ⁴ on Armature draw types and Armature envelopes.

1 Chapter 153 on page 1159

2 Chapter 154 on page 1171

3 Chapter 155 on page 1183

4 http://www.blender.org/cms/Blender_2_40_alpha.598.0.html

153. Object View

153.1. The Armature Object

Armature Object is like any other object type:

- It has a center, a position, a rotation and a scale factor.
- It can be edited.
- It can be linked to other scenes, and the same armature data can be reused on multiple objects.
- All animation you do in object mode is only working on the object, not the armature's contents like bones.

Try it now: add an armature to your scene: SPACEKEY --> Add --> Armature.

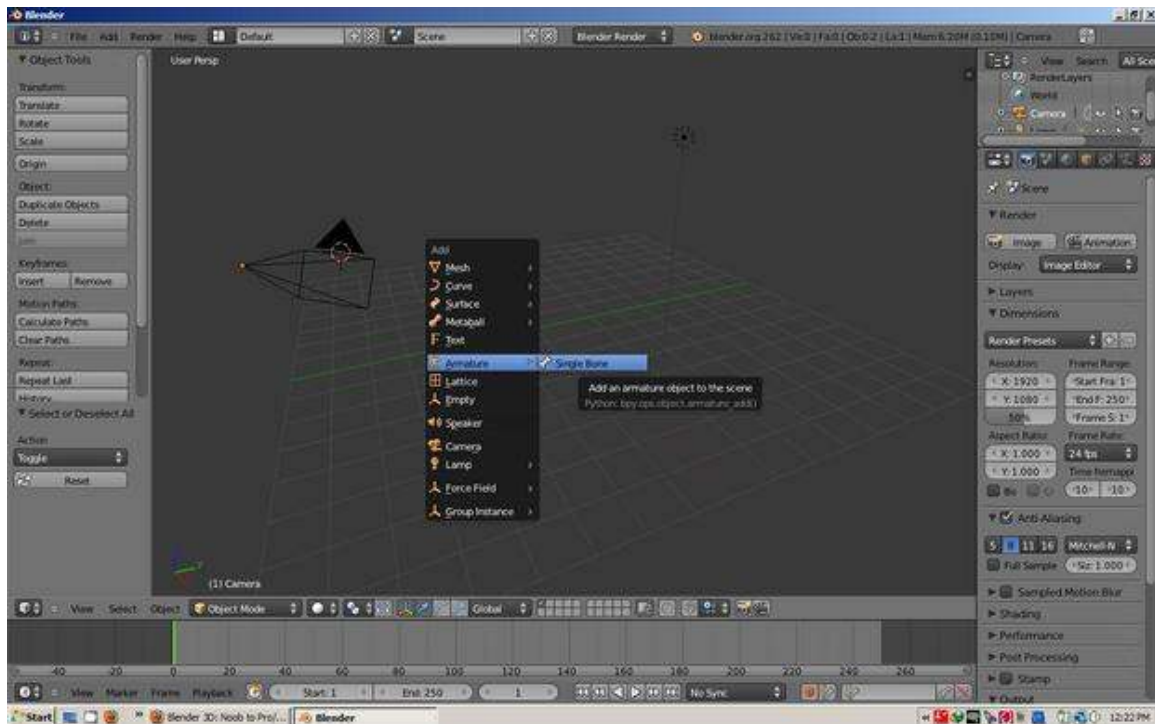


Figure 995 How to add an armature bone in Blender 2.62

Or you can press the Shift + A keys, and select single bone from the armature menu.

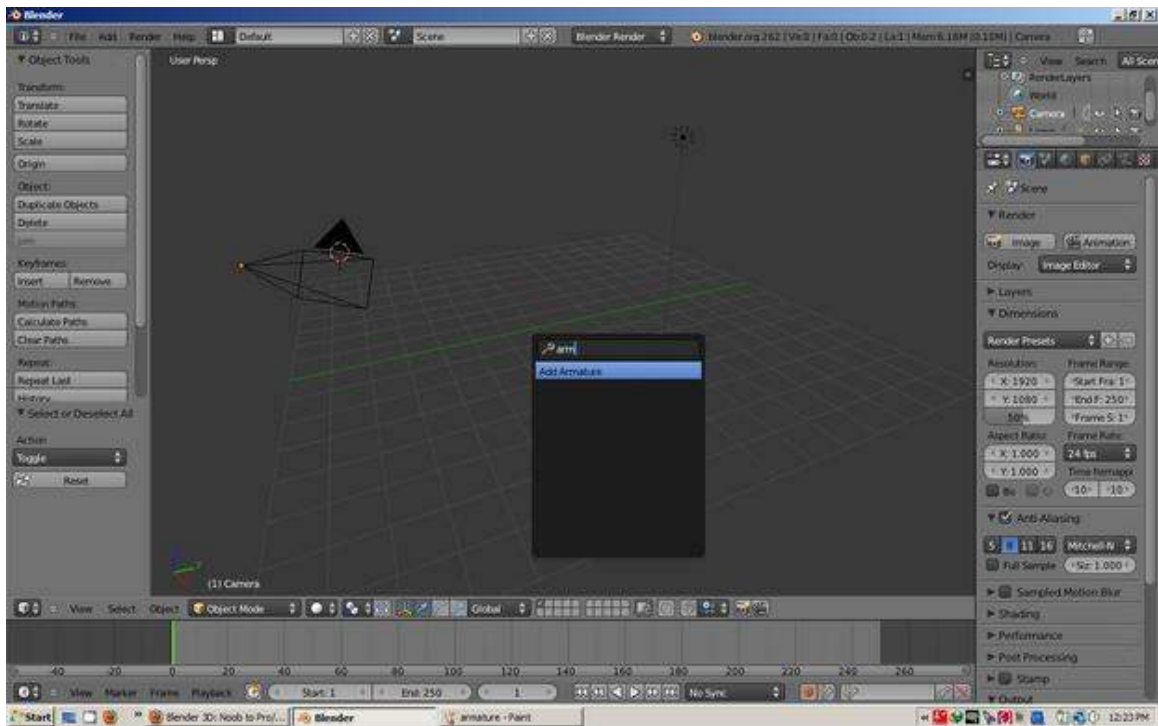


Figure 996 How to add an armature bone in Blender 2.62

When you add a new armature, you'll enter editmode automatically. To switch between modes, use the TABKEY or the dropdown menu in the Header of the 3Dview window (This might not be the case with blender 2.49b, you won't enter edit mode in that version):

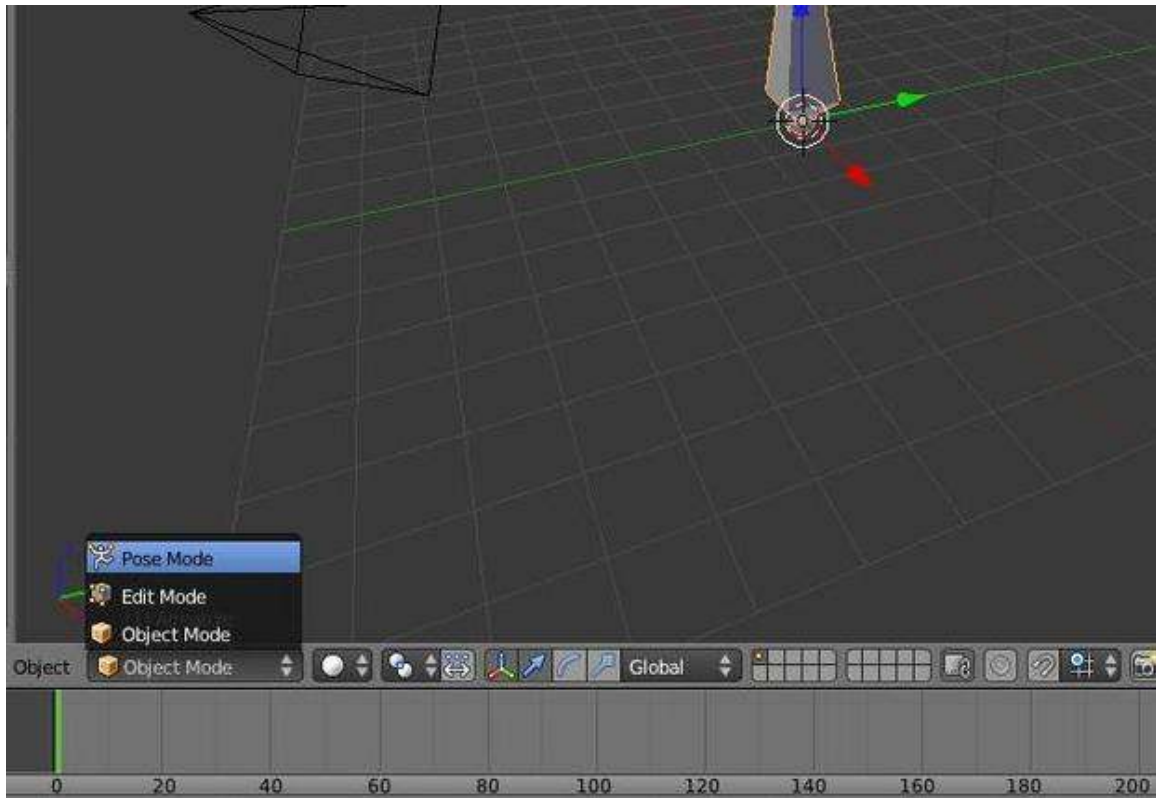


Figure 997 Modes that relate to Armature bones

153.2. The Edit Panel When in Object Mode

This is how the edit panel looks after you have added a new armature and switched to object mode (TABKEY):

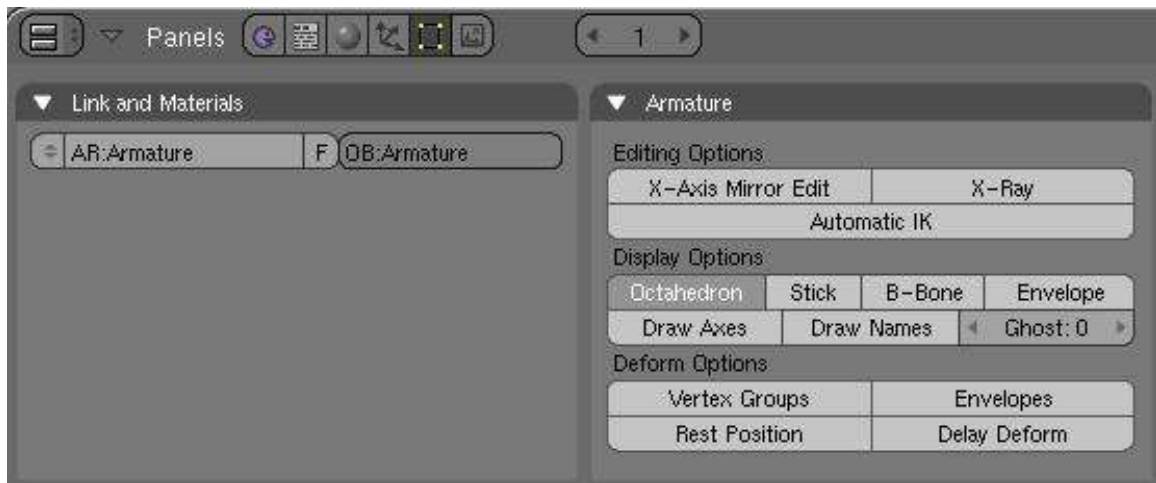


Figure 998

- Link and Materials panel:
 - The AR: field let you rename your armature Datablock. The dropdown is a quick way to select which Armature datablock you want to connect to this armature. You can keep more than one version for the same character. Useful when you have a special move to achieve in a shot, you can turn on an armature for a special purpose.
 - The F button is an option to assign a Fake user to the Armature. Again if you have more than one armature for your character, it's a good idea to turn the Fake on, because if your armature datablock is not used (linked) it's not going to be saved in your .blend files. You can always do batch Fake-assignment of armatures by opening the Datablock browser (SHIFT-F4KEY), go in Armature datablock, select all the armatures you want to keep, and Press the FKEY.
 - The OB: field is just to Rename your armature Object to something more cool and useful than Armature... Armature.001...
- Armature panel:
 - Editing Options:
 - X-Axis Mirror Edit: Not really useful now, it's more of an editmode option. This feature tells Blender you want to replicate all of your bones on one part of the Armature to the other. It's a clean way to just do half the job ;). The axis of mirroring is X so left<-->right in frontview (NUMPAD_1KEY) and the center is the center of the armature object. We will see this feature in detail in the next page.
 - X-Ray: This option will let you see the armature through anything in the scene, solid or not. It's useful to see where your bones are in your character so you can select them.
 - Automatic IK is a Posemode option. It lets you pose a chain of bones as if the bone you were holding was an ik target. More info in Posemode page.
 - Display Options: These options give you the chance to visualise your bones in various ways. Also note that there is some specific options and features regarding the display mode you're in.
 - Octahedron: This is the default view. Nothing exciting except you have a good idea of the rolling of the bones.

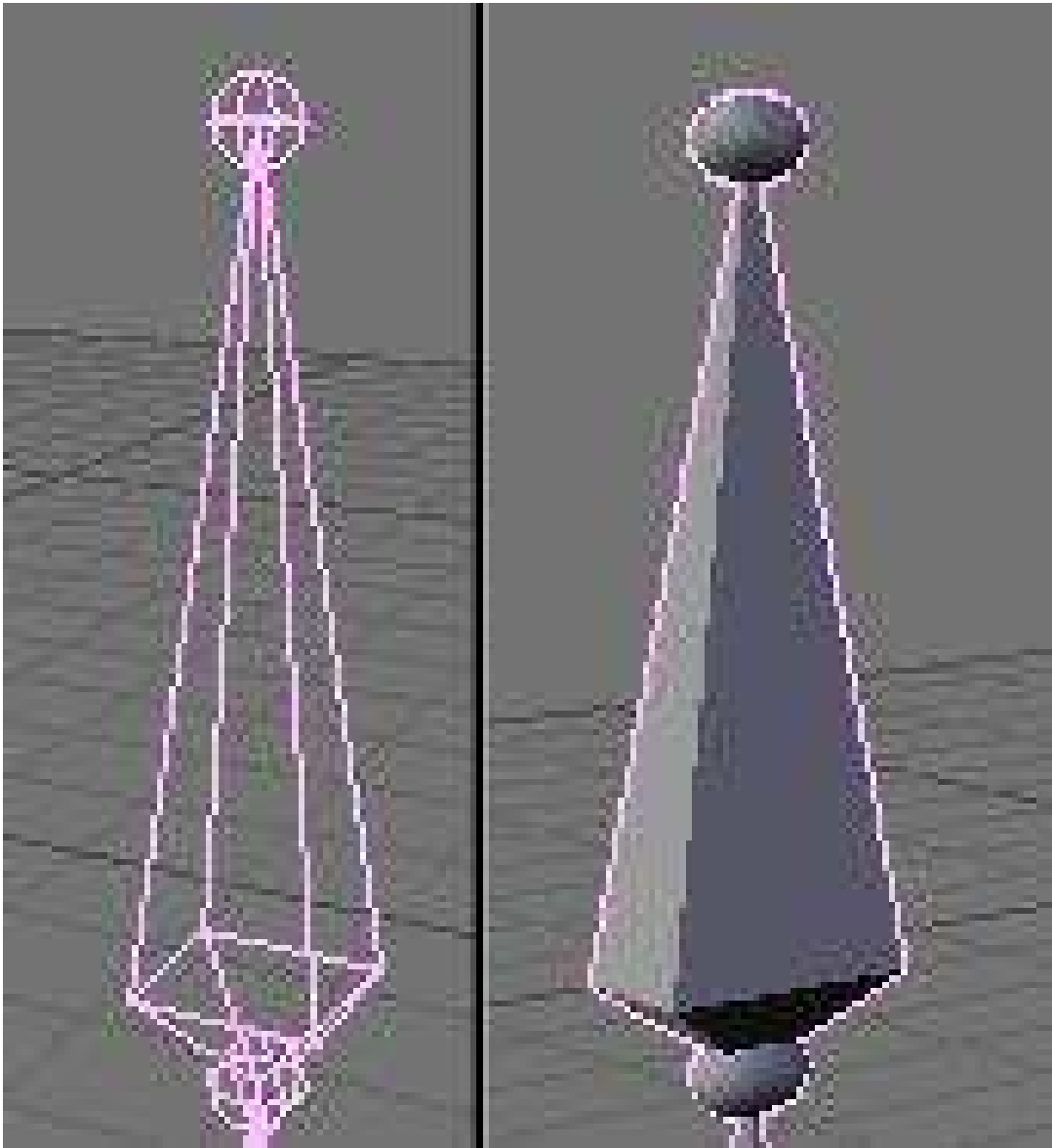


Figure 999

- Stick: This display mode is really useful when you have a lot of bones in your view. It lets you "unclutter" the screen a bit. It draws the bones as tiny sticks.

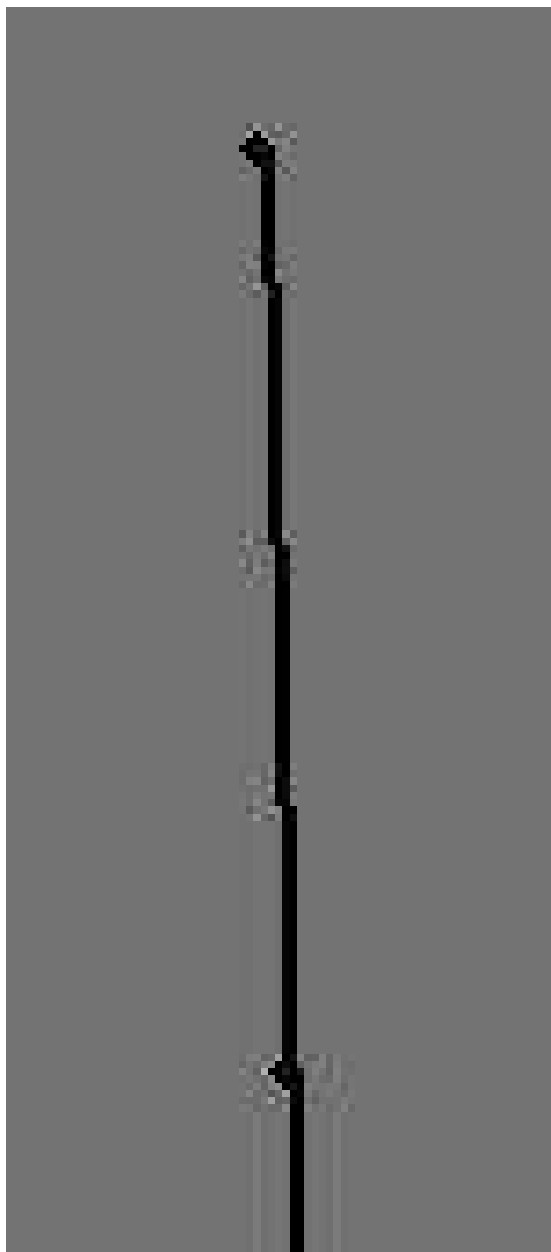


Figure 1000

- B-Bones: It's more a feature than a display mode. This is only useful to visualise the effect you get when you activate the B-bones (Bézier-Bones). Each bone acts like a curve handle and lets you get extremely curvy poses. This will be exposed in the following pages.

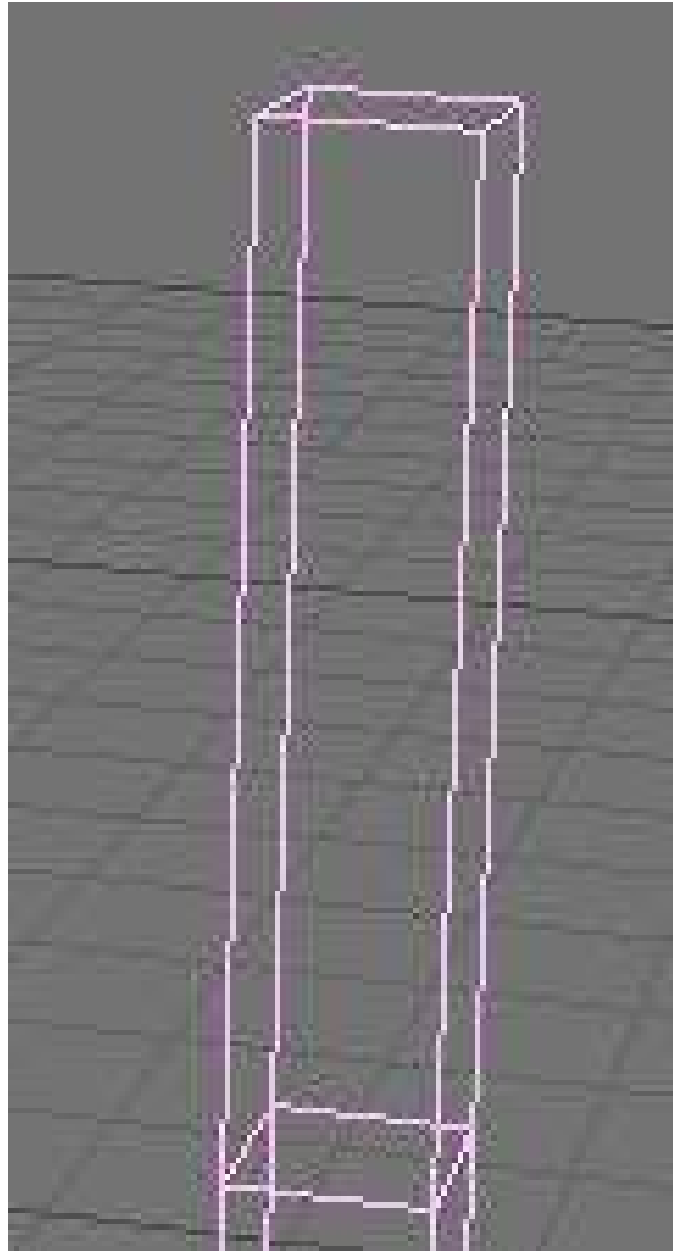


Figure 1001

- Envelope: Again it's more a feature than a display mode. But in this case the visualisation will be useful to tweak your rig later. Envelope lets you easily tell which part of your character this bone will animate and it's visually possible to change the zone of influence exclusively in this display mode. The zone is only visible in Editmode or Posemode though.

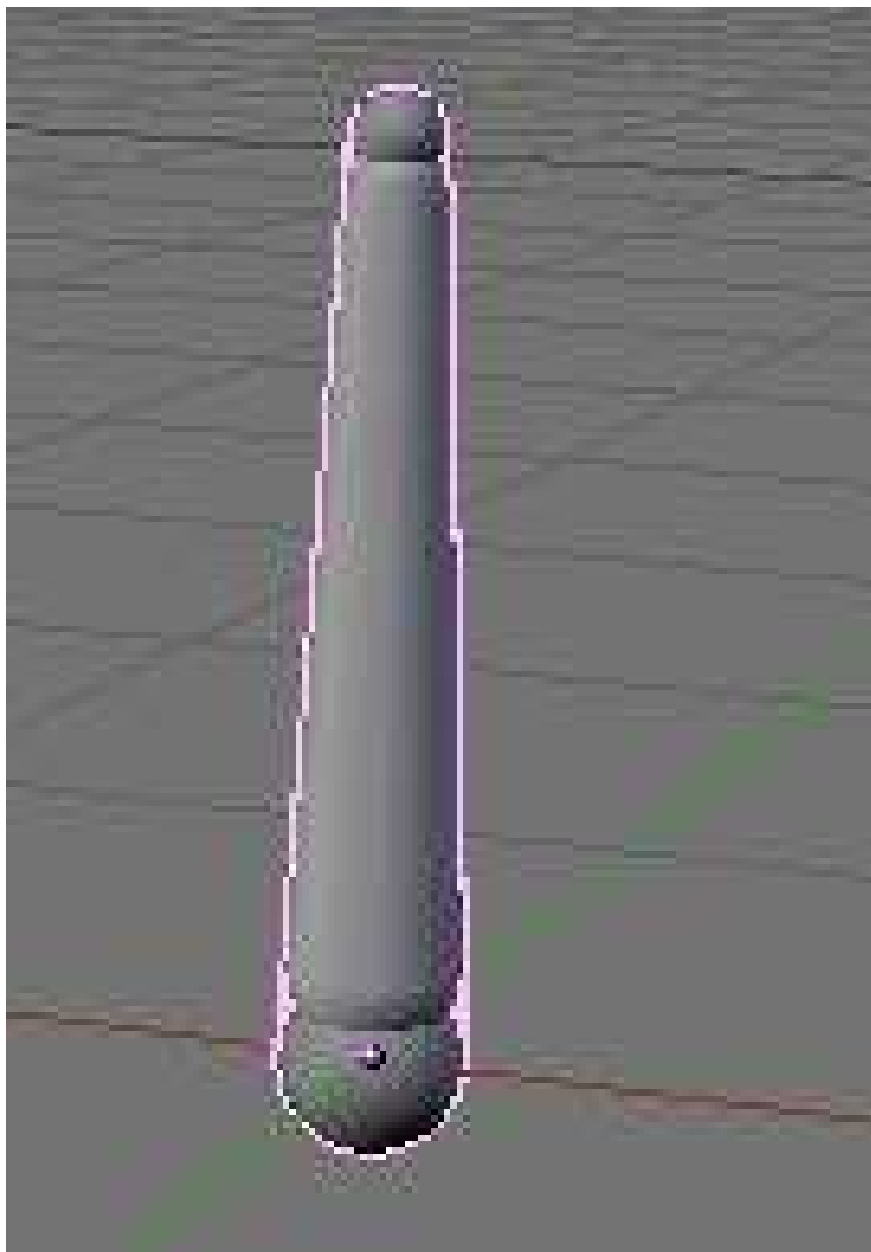


Figure 1002

- Draw Axes: To draw the axes on each bone of the armature when you are in Editmode or Posemode. Useful when you want to know where you are and which axis to use in a constraint for example. Mental note: Y is up, Z is depth and X is side, contrary to object for which Z is up, Y is depth and X is side.

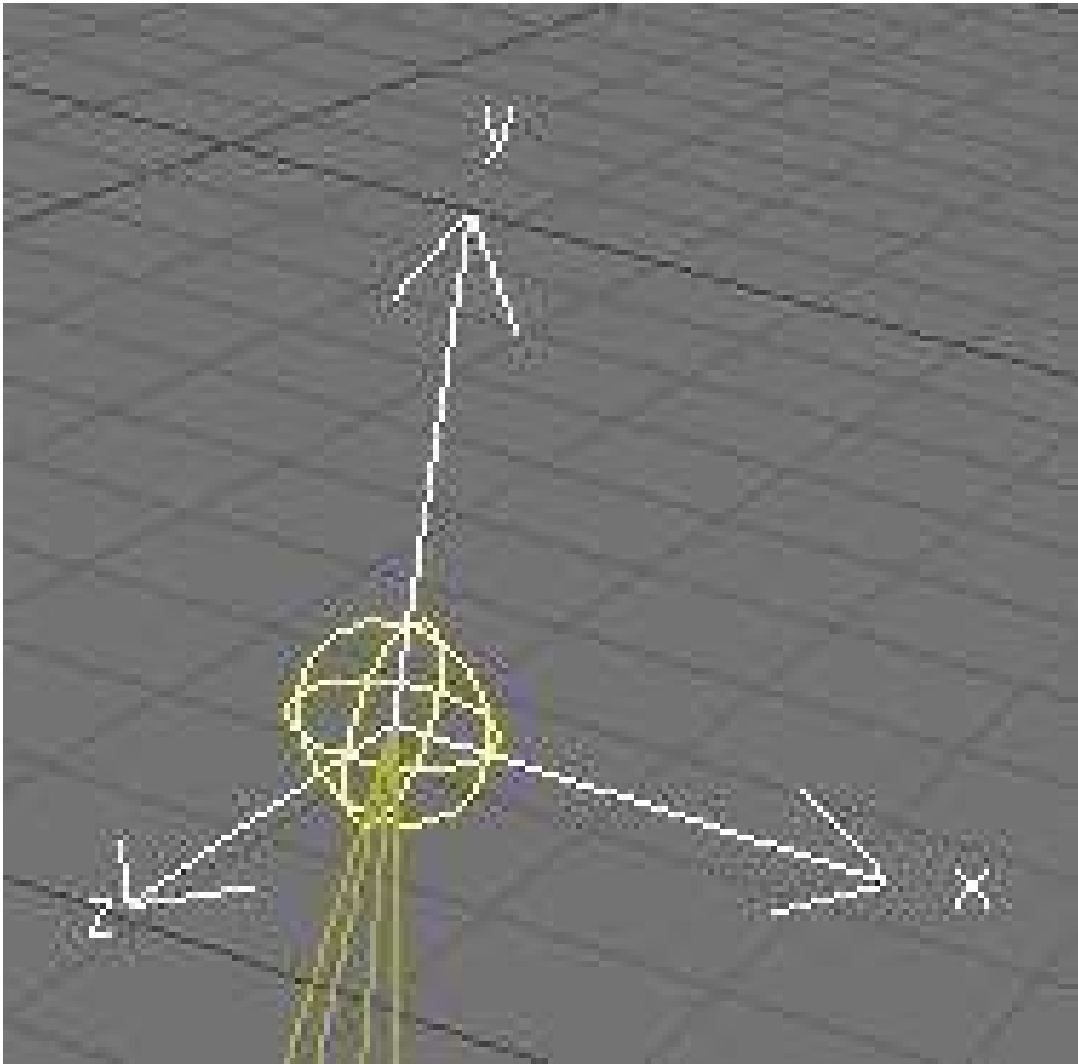


Figure 1003

- Draw names: This lets you see names of bones whatever the mode you are in. It's useful again to edit your armature, create parent dependencies or add constraints.

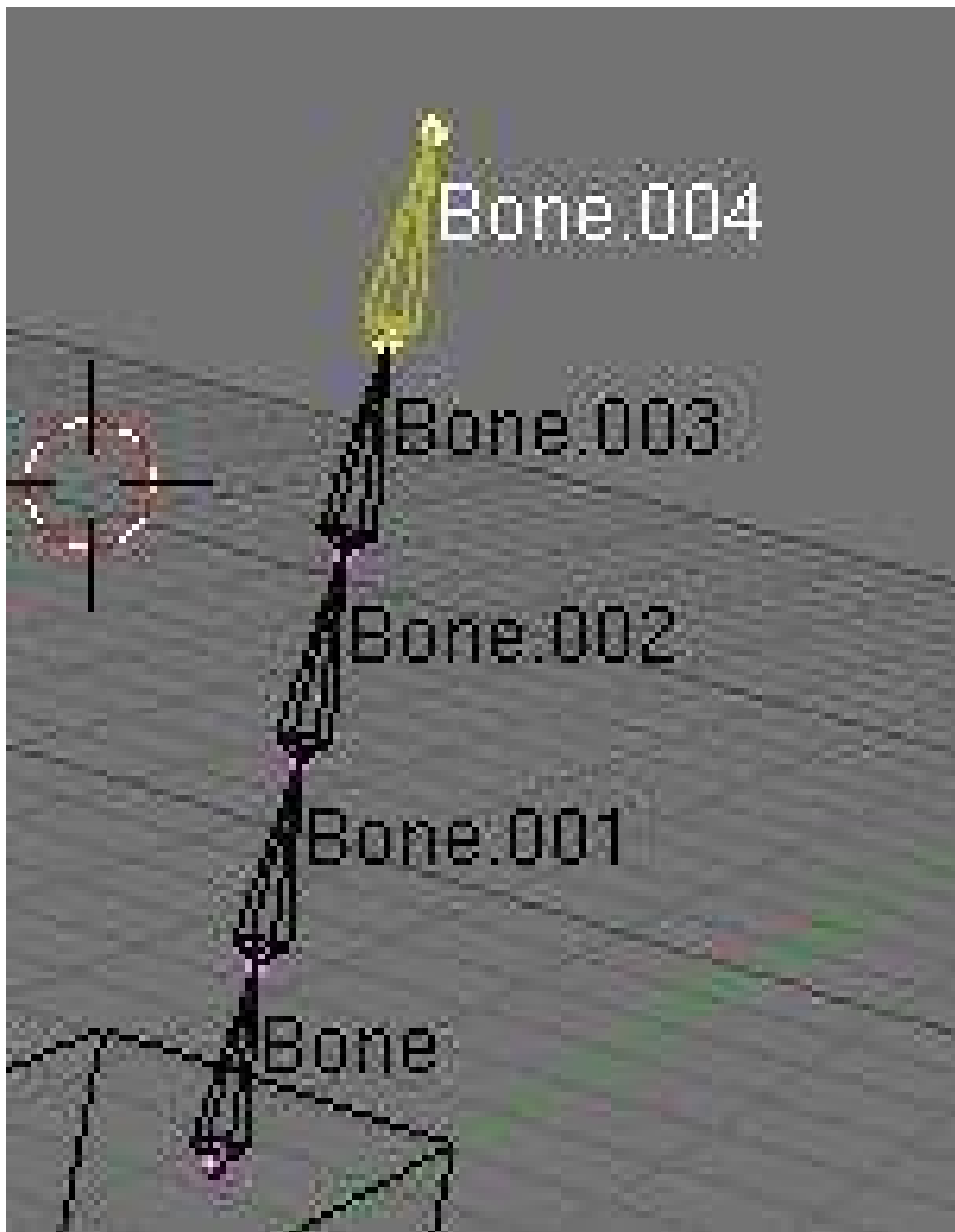


Figure 1004

- Ghost: This option lets you see a ghost of the armature in frames behind and over the current time. This is only working when you have an action linked to the armature, as we will see later.
- Step (Armature_button_obj.jpg needs update): This option lets you choose the frames interval between ghost instances.
- Deform options:
 - Vertex Groups & Envelope: These two toggles let you choose if you want the armature to deform your character using the Vertex Groups and/or the Envelopes. We will see that later.
 - Rest position: This will bring the character back to factory default (item as Editmode), and no actions will be applied to the armature so you can easily edit it in the middle of an animation.

- Delay Deform: This was useful before as the old system was *very* slow. What it does is when you do a manipulation to the rig, it waits until you finish to update the view. Can still be useful though.

153.3. Extra Practice

This YouTube tutorial might also help: [Link¹](#) The short tutorial might be a help [Link²](#)

1 <http://youtube.com/watch?v=GbFvri1ssNo>

2 <http://www.youtube.com/watch?v=T1-tvJVokr8>

154. Edit View

Now you've got your armature¹, but it's not much use until you add some more bones to it. Think about your body for a moment -- you've got this thing you call a 'skeleton', which for our purposes corresponds more or less to an armature object. Your skeleton consists of a number of bones (about 206, to be precise), but generally these are not independent from each other. If you move your femur (the bit of your leg between your pelvis and your knee) then conveniently the rest of your leg moves with it. In that example, the tibia/fibula would probably be counted as one bone, with the femur as their 'parent' bone. In this way, you build up a hierarchy of bones, making animation much simpler.

Editing an Armature Object gives you the chance to add, move or connect bones together. Whilst in edit mode, you will see all of the bones within the currently selected Armature.

When you create a new armature in Object mode a single bone will automatically be created for you, centered at the cursor position. Blender will then switch straight to Edit mode to allow you to add further bones. At this point we're just defining the default 'rest' position of the bones and specifying how they connect together -- you'll have to wait until the next chapter² to find out how to create specific poses.

1 Chapter 153 on page 1159

2 Chapter 155 on page 1183

154.1. Now the basics about bones

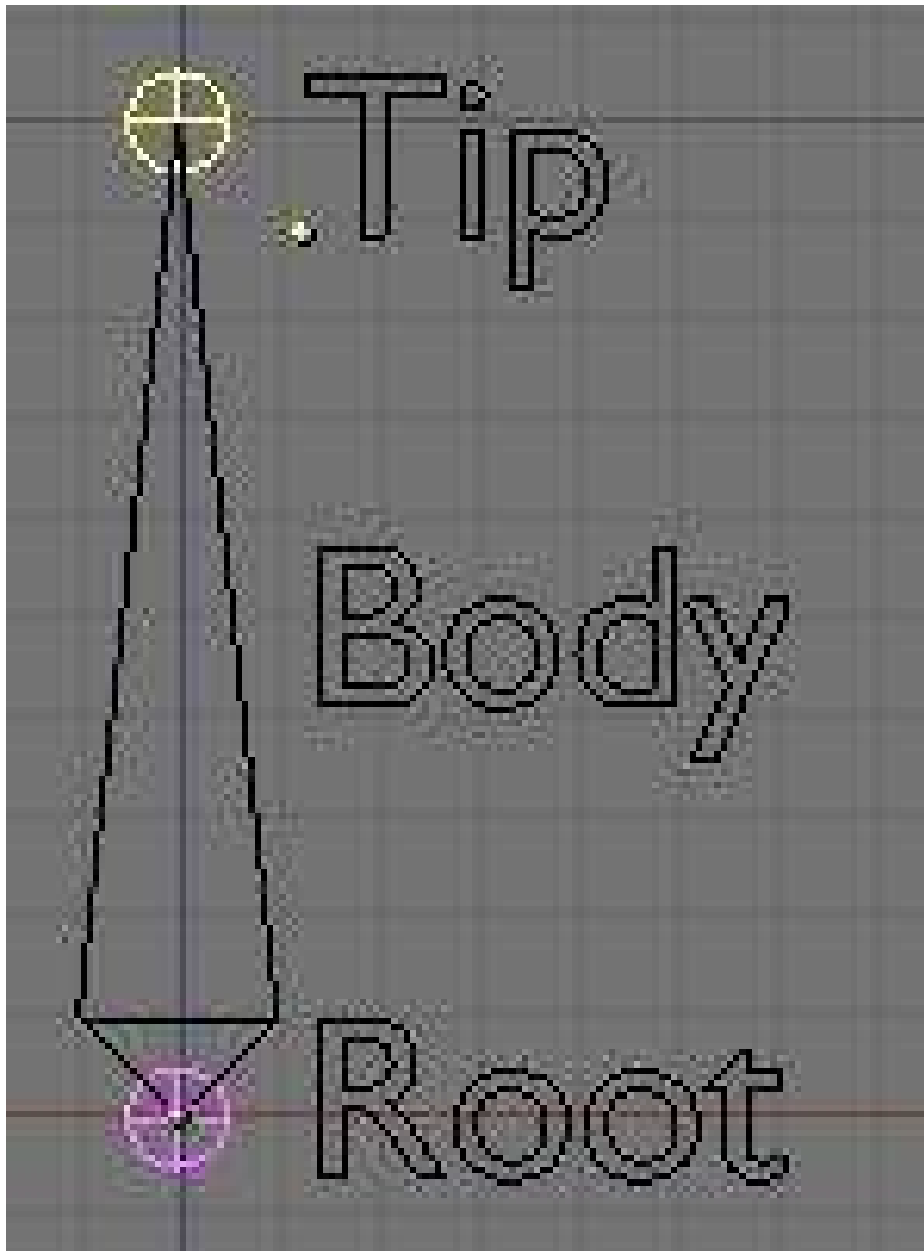


Figure 1005

Having created and selected an armature in Object mode, you can add and modify the bones in this armature by switching to Edit mode.

- You can add a new bone at cursor position by pressing SPACEKEY in the 3DView --> Add --> Armature.
- A bone has two ends: a root (the lower part) and a tip (the upper part). You can select and move the tip or the root independently with RMB, or you can select the entire bone by clicking on its body.

- You can extrude a new bone from the selection using EKEY. This will create a bone connected to the original one, meaning the Root of the new bone will follow the Tip of the original one. You can also CTRL-LMB to extrude a new bone. It will extrude to where you clicked.
- Alternatively, you can connect two existing bones by selecting them one after the other and pressing CTRL-PKEY. You can then choose either 'Connected' (the child bone - the one you selected first - will automatically be moved so that it touches the parent) or 'Keep offset'.
- You can use SHIFT-DKEY to duplicate a bone
- Using the WKEY menu, You can subdivide your bone or flip the name of the bone between Left-Right (See Naming convention below).
- You can delete the bone with XKEY
- You can select a chain of bones (connected together) using LKEY, when you hover your mouse over a bone.

154.2. The edit panel

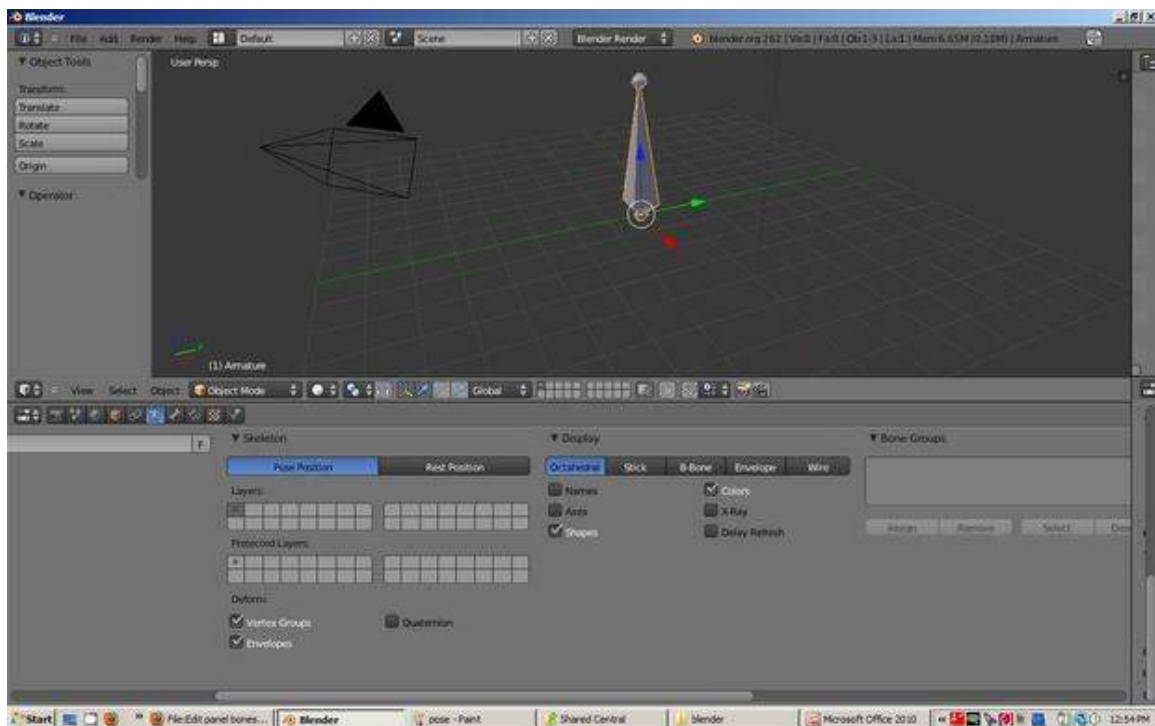


Figure 1006

- Armature Bones Panel
 - OB: this field lets you rename your bone.
 - "Child of" Dropdown: lets you choose which bone will be the parent of this bone. If a parent is selected, there will be a small button labelled "con", meaning connected. Setting the parent defines the relationship between your bones. When one bone has another as its parent, it will do everything the parent does, such as rotate, move and scale. A dotted line between the parent and child will appear. If you select Connected, the Root of the Children will go stick to the tip of the parent, giving you a chain of bones like the 2 bones in your arm.

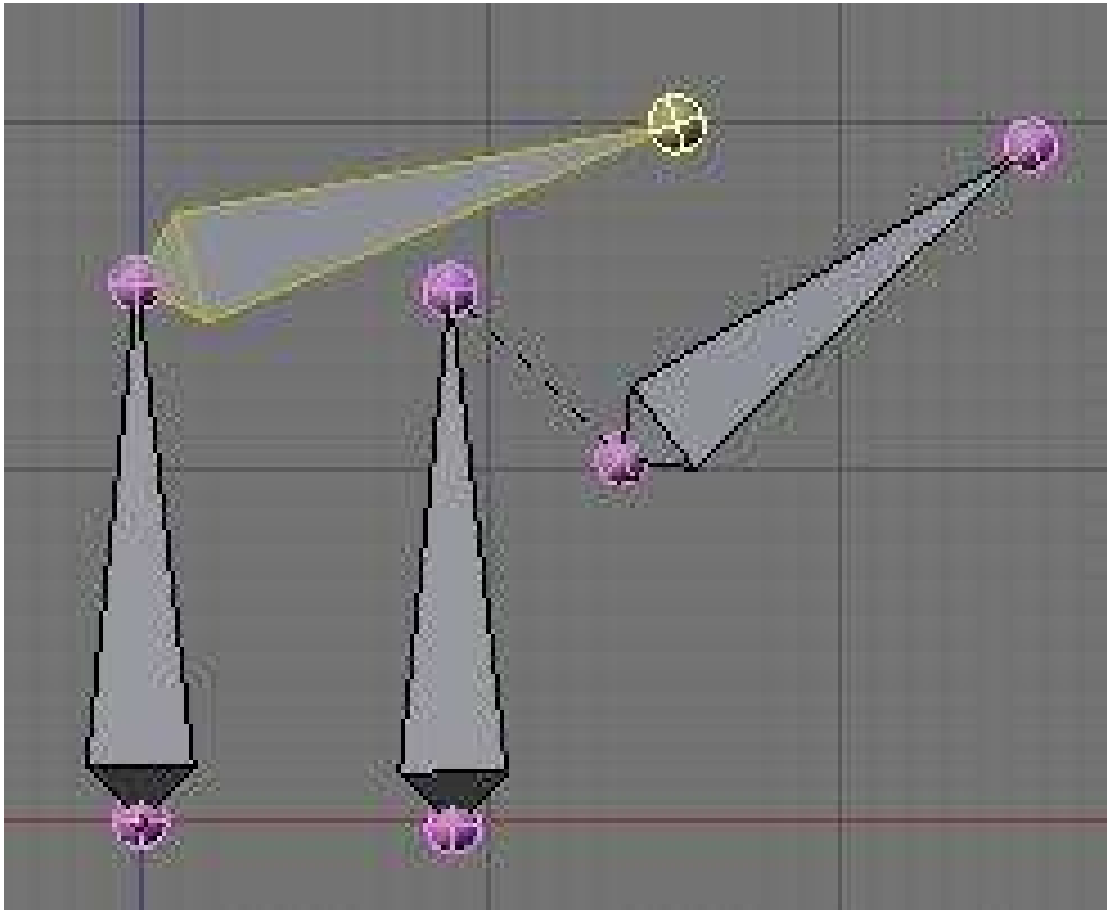


Figure 1007 none

- Segm: If you set this value to something greater than 1, it will cut your bone into several little segments and deform them on a bezier curve - referred to as a 'B-Bone'. You need to create a chain of bones to really show off this feature though. In the example below, the image on the right has 1 segment, and the one on the left has 3 segments each (these are shown in Object mode to show the effect more clearly):

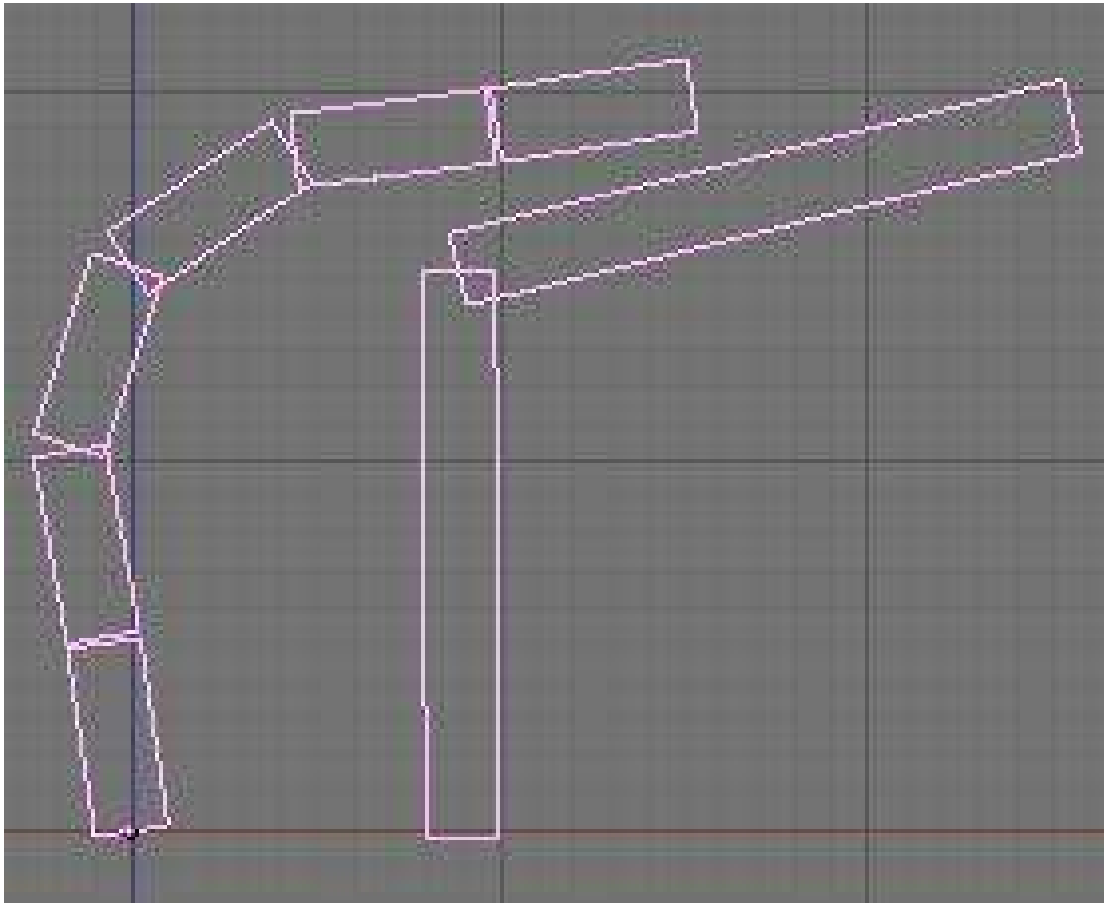


Figure 1008 none

- **Dist:** This is the area of influence of the bone. It can be visualised using the Envelope³ display mode. We generally don't touch this field as there is an easier and faster way to change this option. Turn Envelope on and select a bone. Then using ALT-S, you can scale the zone of influence. This has the advantage that you can do it on multiple bones simultaneously, and it works in both editmode and posemode:

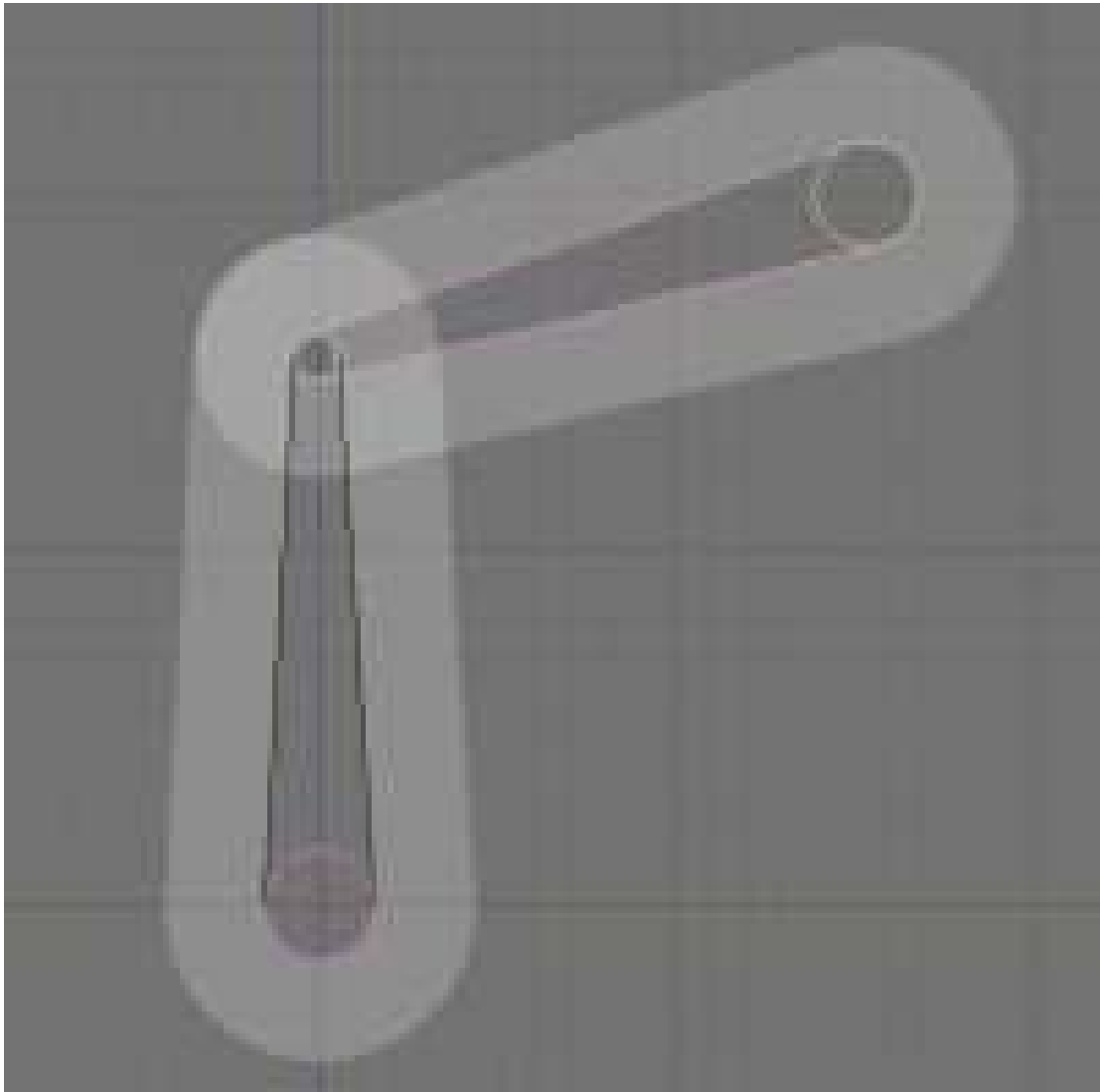


Figure 1009 none

- **Weight:** This specifies how strongly this bone will influence the geometry around it, relative to the other bones. If two bones crossing each other, both with envelope influence, have the same weight (like 1:1) they will influence the surrounding geometry equally. But if you set one to 0.5, the geometry will be affected more significantly by the other one, with weight 1. For example, in this image, 2 bones using envelope influence try to move the same geometry. The 2 on the left have the same weight, you can see the geometry didn't move. On the right, one of the bones has 0.5 so the bone with weight 1 is winning the tug-of-war!:

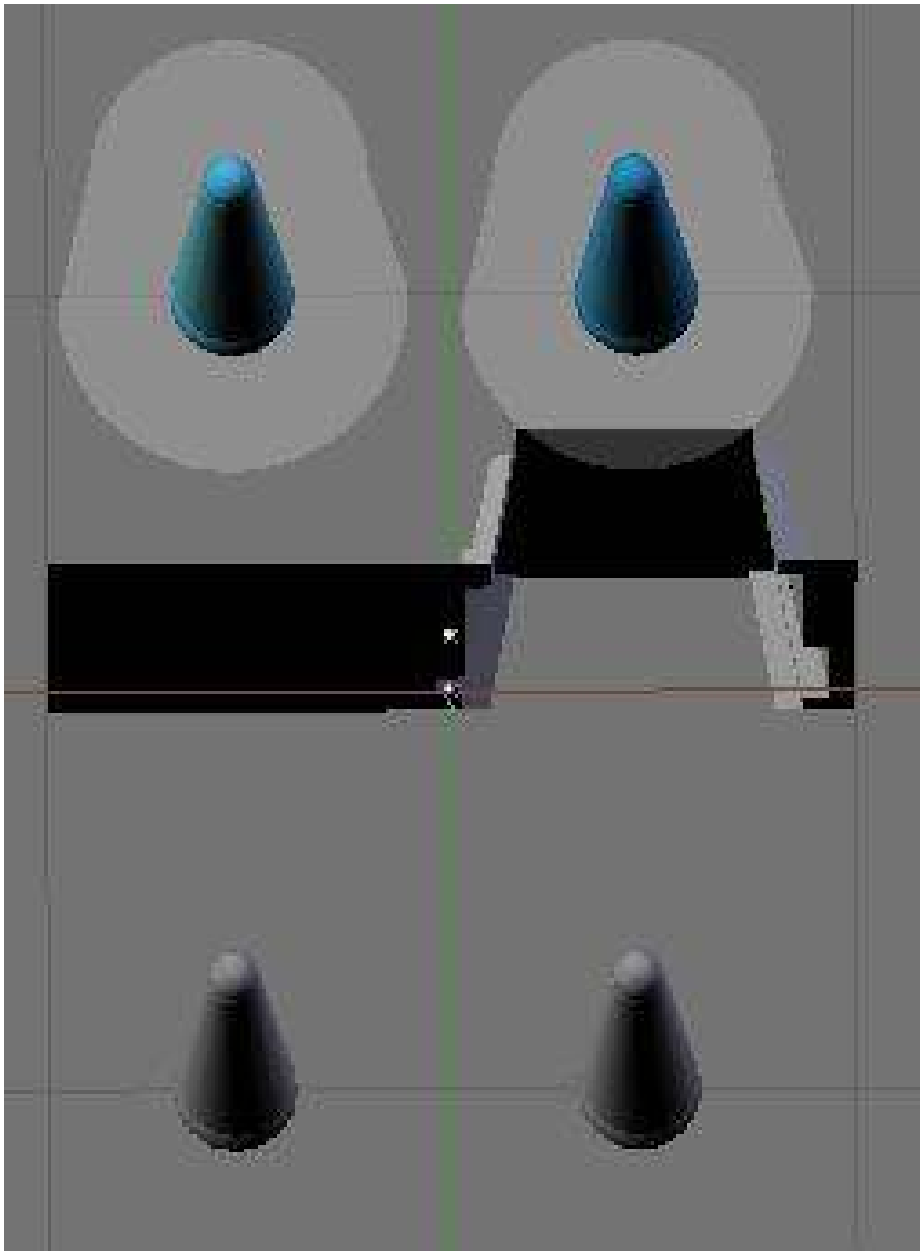


Figure 1010 none

- Hinge: This tells the bone to remain motionless in a chain. It doesn't copy the rotation and scale of the parent. Useful for mechanical rig I would say, as you can animate the rotation of the hinge bone without having to correct it because the parent rotated:

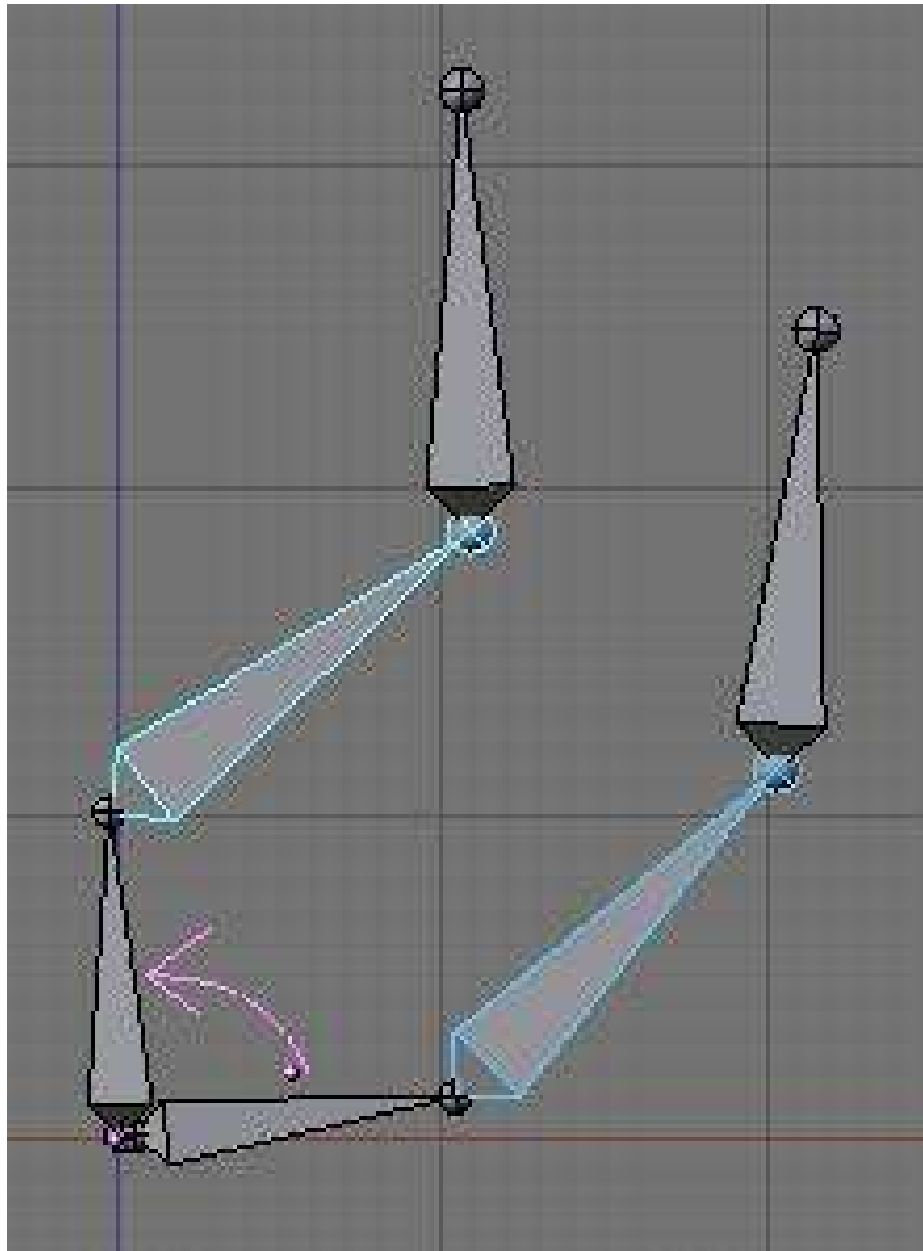


Figure 1011 none

- **Deform:** This lets you say if you want the bone to deform the geometry at all. Switching it off is like setting the weight to 0, except it's faster this way. Useful when using a bone as a target or a controller, i.e. a bone you just want to use to control other bones, but not the geometry itself.
- **Mult:** to deform geometry you can use vertex group⁴ and/or Envelope⁵. The ability to mix both of these methods is handy for using one to tweak the other. For example, you might use envelope everywhere but tweak difficult places manually with vertex group. We'll discuss this in more detail later on.

4 Chapter 159 on page 1199

5 Chapter 158 on page 1193

- **Hide:** This option lets you hide the bone. You can use it to hide the less important bones when you want to see what you're doing or for when you come to animate later on. For example, when you animate you don't need to see the entire chain of the leg, just the controllers. The values you select here apply to both Editmode and Posemode.

154.3. Naming convention

In many cases, rigs are symmetrical and can be mirrored in half. In these cases, it is helpful to use a left-right naming convention. This is not only useful for your own sake, but it gives Blender a hint that there is a pair of equivalent bones, and enables the use of some very cool tools that will save you some significant work.

- It's helpful to name your bones with something useful telling you what it's there for, such as leg, arm, finger, back, foot, etc.
- If you get a bone that has a copy on the other side, however, like the arm (you have 2 arms right?), then the convention is to call them arm.Left and arm.Right.
- Other alternatives are also possible, like `_L`, `_LEFT`, `_left`, `.L`, and `.Left`. Anyway, when you rig try to keep this left-right thing as accurate as possible; it will pay off later on.
- You can copy a bone named `blah.L` and flip it over using `WKEY` --> flip name. So the bone will be `blah.L.001` after you copy it, and flipping the name will give you `blah.R`. Blender handily detects if the `.001` version already exists, and increments the number for you.

This is an example of naming in a simple rig:

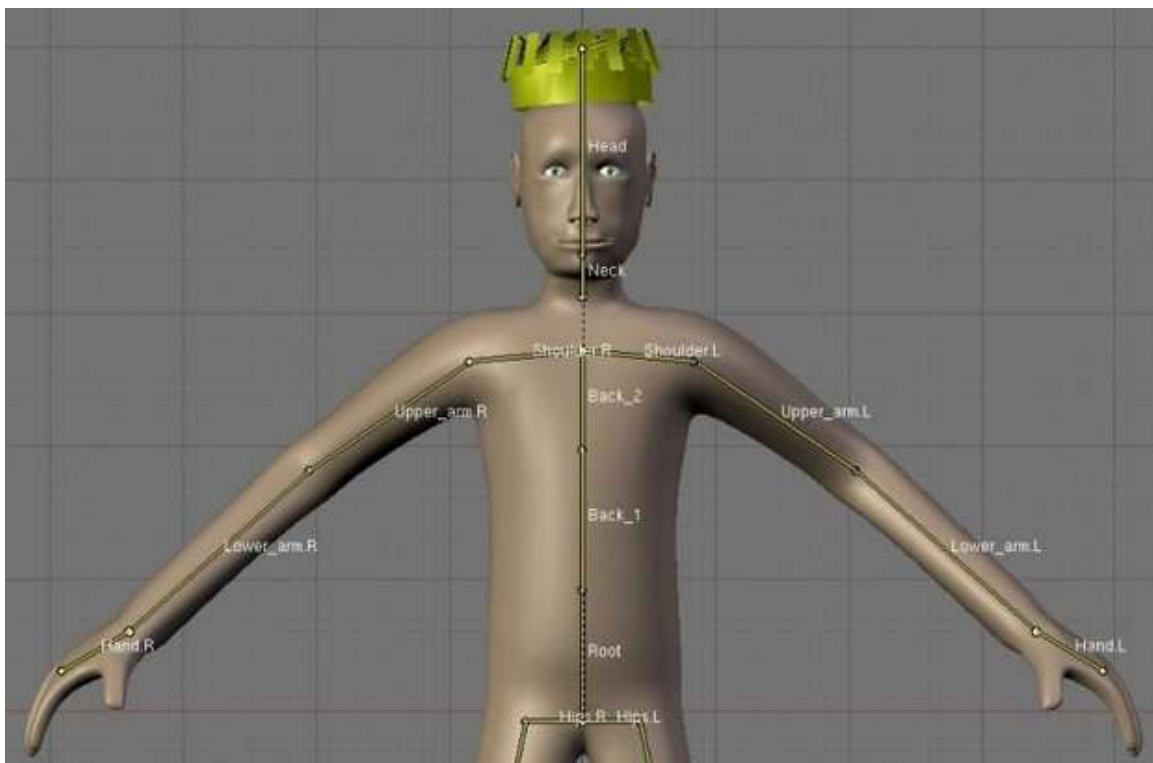


Figure 1012

154.4. Mirror Editing

Now we come to the X-Axis Mirror Edit feature. This handy little number allows you to define only half of your character and tell Blender to automatically repeat the same actions on the other side. It's cool, it's simple and it saves a whole lot of time.

We will create a little guy out of sticks for the occasion -- don't worry about the geometry yet.

- Create a new, empty scene. Since the mirror editing feature mirrors along the X Axis, make sure you've got the front view selected (NUMPAD_1KEY) so that the X Axis runs from left to right. Add a new armature to the scene (SPACE, AddArmature). Enable 'Draw names' from the 'Display options' section of the Editbutton panel, so we can see what we're doing.
- Now enable mirror editing by pressing F9 on the buttons window and clicking the **X-Axis Mirror** button in the **Armature** panel under **Editing options**. You'll also need to use the center of the armature (indicated by a purple dot) as the center of your rig, otherwise the symmetry will go wrong when we come to create the mirror image.
- Name the first bone you have "Back". You can scale it to make the entire back of the guy.
- Select the tip of this and extrude a new bone from it to do the Head. Name it Head.
- Select the tip of Back again and do **SHIFT-EKEY** to tell Blender you're starting a **mirrored** chain of bones. Blender will automatically extrude another bone and will create an exact mirror of whatever you do. Take note that the name of both bones are Back_L and Back_R. Blender also tries to keep to the naming convention. Unfortunately, since we extruded from the Back bone, the names aren't quite right anymore.
- To change the names: Start by editing one of the names as Arm. Add the suffix to it (_L or _R). Then hover your mouse over the name field and do CTRL-CKEY. You just copied the name of the bone! Select the other bone, hover your mouse over the name field and do CTRL-VKEY. This will paste the name as-is. But as there is already a bone with the same name, Blender will add .001 after it. No problem; just go into 3Dview and do WKEY --> Flip name. There you have it -- a working mirror again.
- Mirror editing works using names. If you move a bone named blah_L and there is a bone named blah_R in the same armature, Blender will mirror the move you do to it, so make sure you follow name convention correctly.
- Then we can continue: extrude an other bone to make the lower part of the arm using EKEY or CTRL-LMB. The new set of bones should be arm_L.001 arm_R.001.
- Then we will add the legs. Up till now we have always worked from the tips of the bone. This is easy as Blender understands you want to create children of the selected bone, but to make the legs you need to extrude from the root of "Back". So go ahead, select the root of "Back" and do SHIFT-EKEY to start a pair of chains. Rename them to "leg"+suffix.
- Now take note that doing so will not parent or connect the new bones to anything. We don't want it to be connected to the tip of "Back", it would look silly. But we want it to follow the body!
- The way to go is to parent the two legs we just created to the "Back" bone. The old way (pre 2.40) was to select all bone and select the parent manually in the drop down. In the new version, editmode and posemode have an *active bone*. The active bone is the last bone you selected. In this case we can't work with more than 2 bones selected. Select the child (a leg) then select the parent (Back) and Do CTRL-PKEY. A menu will popup asking *Connected* or *Keep offset*. For now use *Keep offset*. Do this for the other leg as well.
- it's also possible to remove parent easily. Select any bone you want to remove parent relation from and do ALT-PKEY. A menu will popup asking if you want to clear all or just to unconnect.

Of course you don't need to select the parent and/or the child for this to work since any parent relationship will be cleared. So if you do that on a bone which is parent of 5 bones, then immediately all the children will be parentless.

- Extrude one more time to get a leg with 2 bones.
- Turn on the Stick display mode and enjoy your guy made of sticks!

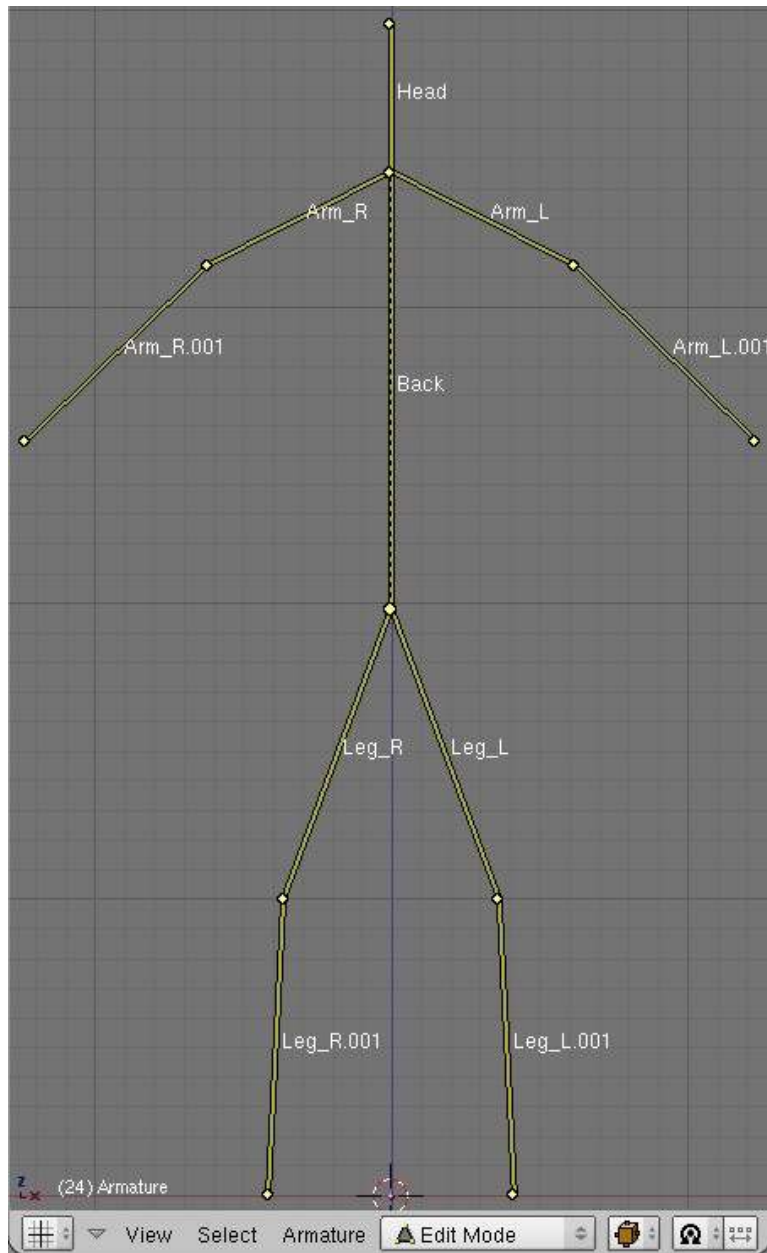


Figure 1013

- Now you can go into Posemode and pose your guy as you want.
- You can move the entire guy just by moving the "Back" bone, since this is how we built him. This bone is the highest in the bone hierarchy, "The daddy of all bones", you could say!

FASTER WAY TO NAME YOUR BONES 1) Create one side of the armature complete with correct bone names like the way this tutorial describes. 2) Copy the side of the armature you just created and paste it on the opposite side to form a complete armature. 3) On the side that you just duplicated, your names would have numbers added to them. Now select each bone on 3D View and hit "W" then select "FLIP NAME". **MAKE SURE YOUR PARENTING IS THE SAME AS THIS TUTORIAL DESCRIBES!**

155. Pose View

Posemode is a very versatile place where you Animate your character, create and manage constraints and apply your rig to your character.

Contrary to Editmode, Pose mode isn't an obligatory mode where you can't do anything else. It's now part of the UI like any other object. A good example of it is you can be in posemode and still select another object.

155.1. So What Can You Do?

When you are done building your armature, you can go into Posemode to add constraints and start creating actions. There are also some new tools accessible in Posemode that you may want to look at. You can easily get into "pose" mode by selecting the mode from IPO type list box in the left portion of the lower screen.

The panel has changed a bit too:

Figure 1014

- What's new in the panels?:
 - You can use the Automatic IK feature in the Editbutton(F9) to pose a chain of bones like it was an ik chain. It's usefulness is very limited though. It works well only if there is no other ik solver in the chain, and if your chain is isolated from the rest of the rig.
 - Ghost: in the armature panel the ghost option lets you see the action linked to the armature over time. Also called onion skinning.

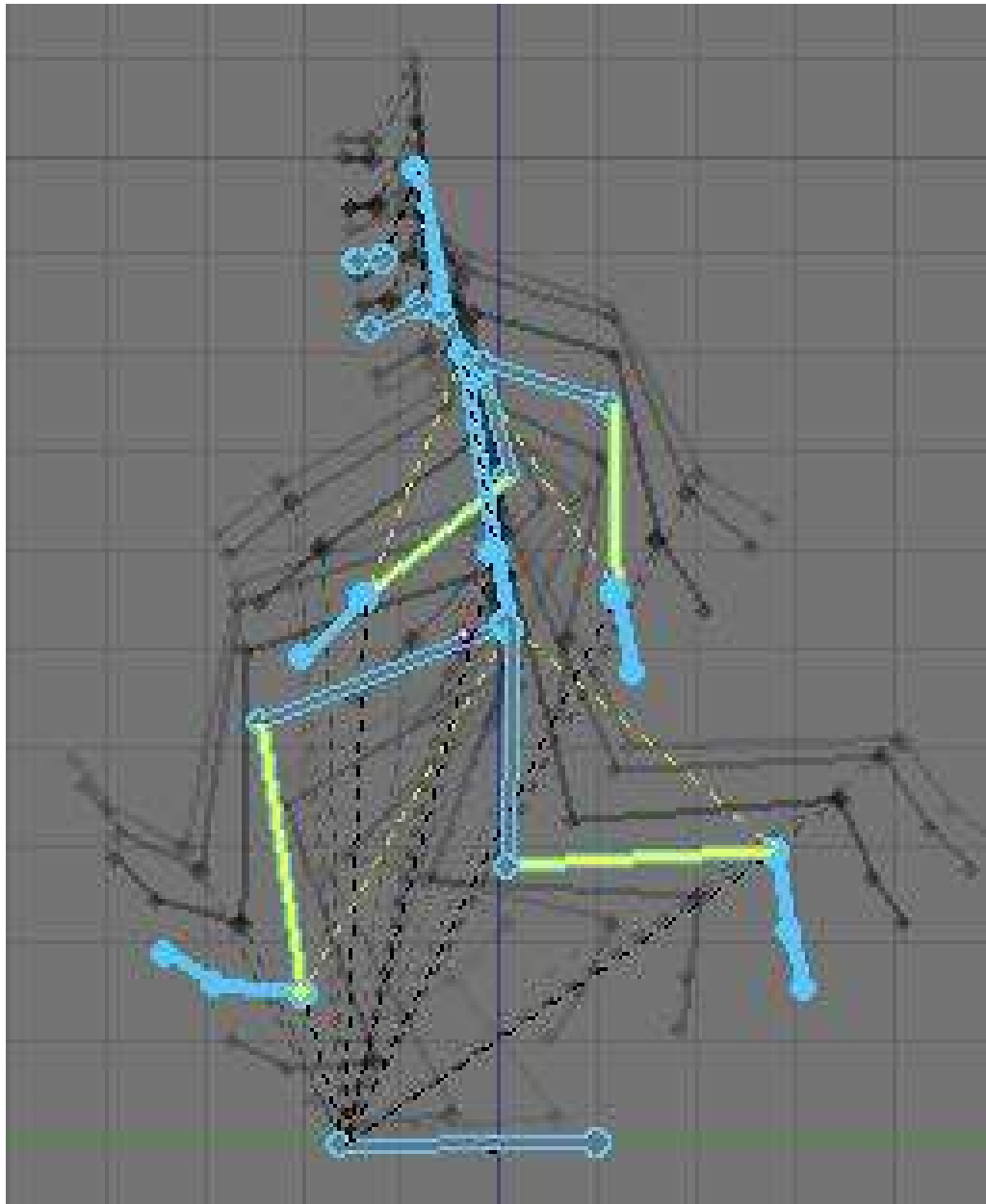


Figure 1015

- There are two number fields to better tweak the effect of B-Bones. The in/out is used to tell the scale of the virtual handle of the bezier curve. In is the Root of the bone, and Out is the Tip. The bigger the value, the bigger the effect of rotation.

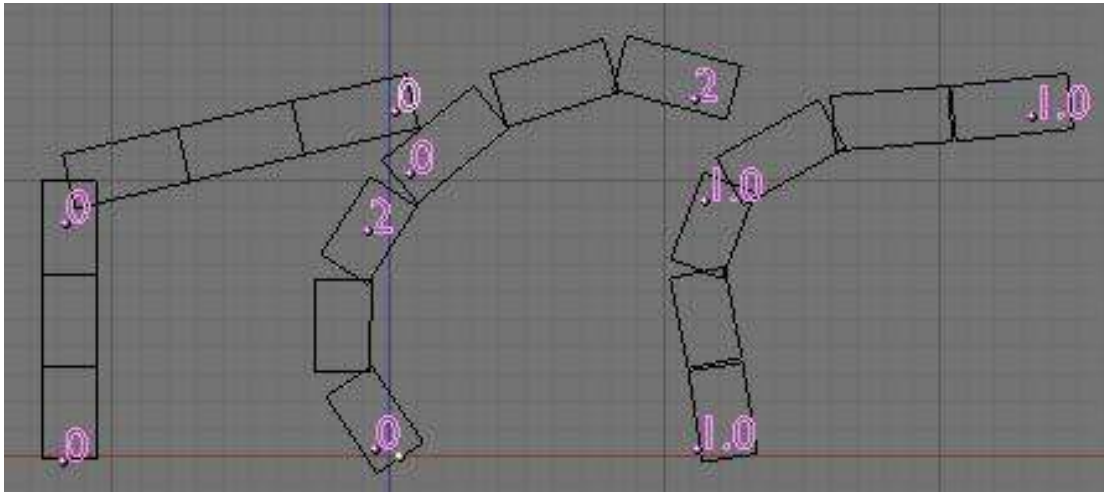


Figure 1016

- There is now a Constraint panel where you can add a constraint to a bone, like any other object in the scene. This will be shown later.
- You can pose your rig using GKEY, SKEY and RKEY. Note that if the bone is part of a chain it can't be moved (except if it's the first of the chain, moving all the chain as they are all children), so you rotate the bone instead.
- You can do ALT-SKEY on one or more bones while in Envelope display mode to tweak the envelope size in real time while animating. Useful when for example you move the hand and some part of the character isn't in the influence zone; the result will be that some vertices will stay behind.
- You can do CTRL-CKEY to copy stuff from a bone to bones. The options are location, rotation, scale and constraint. Constraint is very handy when you want to copy a constraint to other bone. The way it works is easy.
- The W-KEY menu gets some neat options too:
 - Select constraint target: Will select the target of the bone's constraint currently selected.
 - Flip name: Yep, you can flip name in Posemode too.
 - Calculate/Clear path: This is a visual way to see the action linked to your armature. You can select just some bones and ask Blender to show you the paths of the bones.

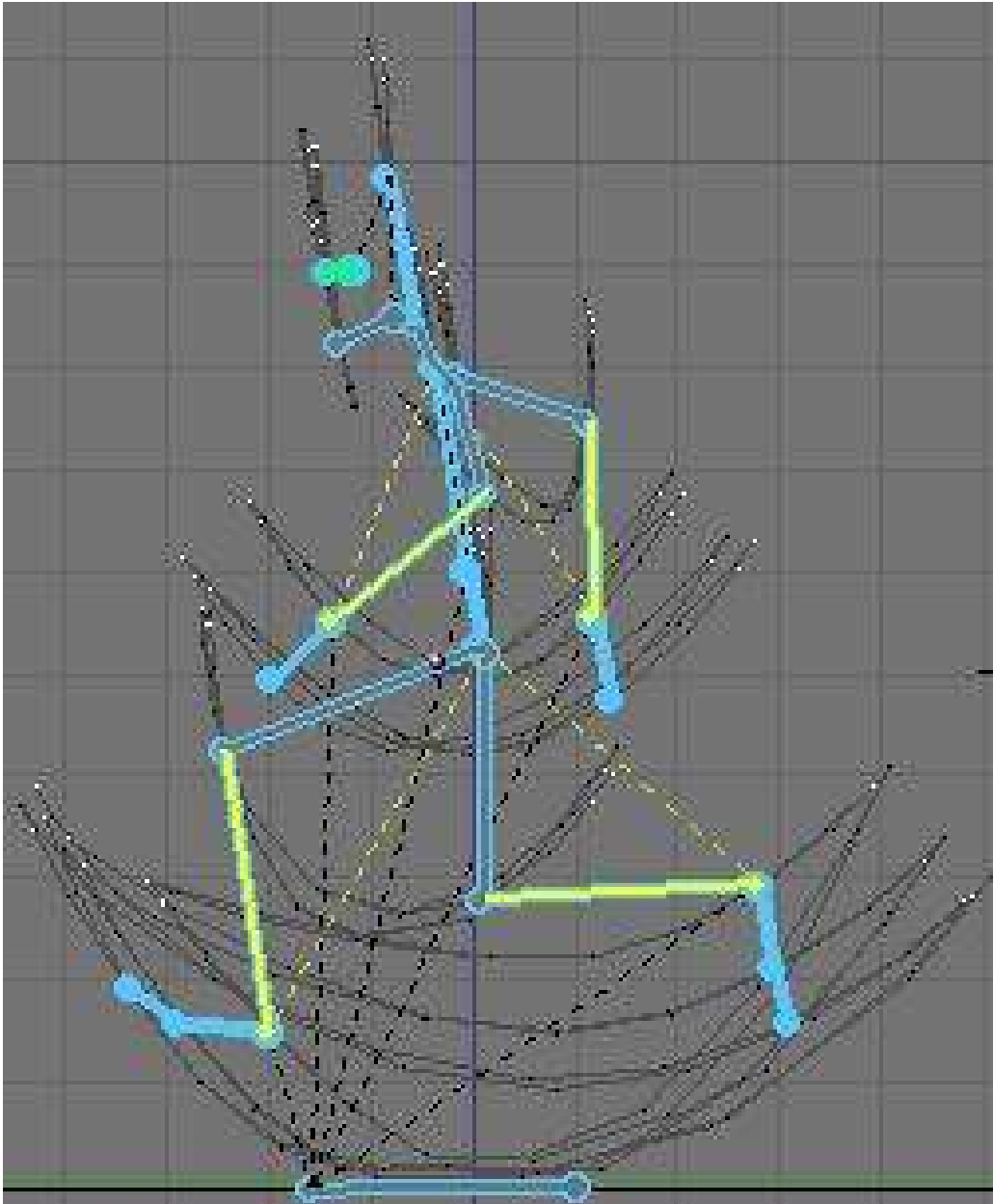


Figure 1017

- You can pose your character and select all bones you want to see included in the action and press I-KEY. You can insert a key just for loc, rot or size. Avail will add a key to all available channels in IPO window (all channels you previously added something).
- When you insert key for your armature, a new action is created and linked to the armature if there was no action before. You can also see the curves of each selected bone of the armature in the IPO window. We will see action window and IPO window later.
- You can parent a bone to an external object by selecting this object then selecting the bone in question so it's active (The armature is in Posemode so you can select a bone). Do CTRL-PKEY. Then when you move the bone the object will follow. This kind of Hard relationship doesn't include any bending at all. It's useful when doing robot rigs as you are just moving objects around.

156. Mesh Object

This section will explain how to deform your mesh using the armature.

There are two ways to tell Blender which vertex will go with which bone: Vertex group, and Envelope.

There is also a tool useful when animating which is part of the mesh object: the Shape key, to create a preset deformation. For example: deform the face to look like a smile.

- [Connection between Armature and Mesh](#)¹
- [Envelope](#)²
- [Vertex Groups & Weight Paint](#)³
- [Shape Key](#)⁴

1 [Chapter 157 on page 1189](#)

2 [Chapter 158 on page 1193](#)

3 [Chapter 159 on page 1199](#)

4 [Chapter 160 on page 1207](#)

157. Armature and Mesh

How to tell Blender: "use this armature to deform this mesh"

157.1. The Armature Modifier

Blender now has a Modifier stack (Editbutton, F9KEY). As such, we should use it over existing methods to pair mesh and armature, as the modifier stack is optimised and simple to use. Note: You don't need to parent the mesh to the Armature anymore. The only case you could need to do this would be animating the Armature object itself. Then the mesh should also follow the armature. In this case select mesh, then armature, and do CTRL-PKEY --> Object.

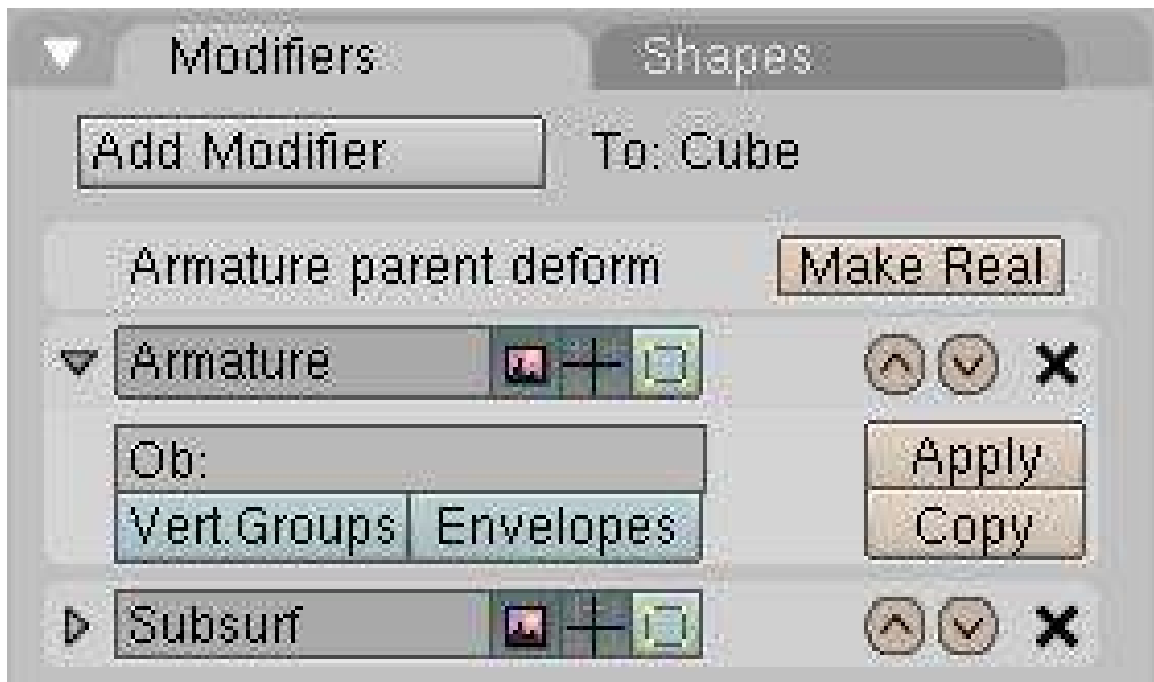


Figure 1018

The clean way to do so is to go in the Editbutton window (F9KEY) and press "Add modifier" in the Modifier panel, then select "armature" in the dropdown menu. Then you'll get a new modifier "Armature" like the previous picture. Change the **OB:** field to the name of the armature object that should control the mesh. This step is very important! Without the armature object being defined, Blender won't know how to modify the mesh since there may be multiple armatures within your world. To limit the effect of the modifier, you can enter the name of a vertex group in the **VGroup:** field. This will minimize unwanted distortions in very dense meshes. There are also fields to enable/disable

the modifier when rendering, enable/disable when working to only move the armature (could get handy with massive character), and when editing (that's very handy, you can edit the topology while it's deformed). There are also two toggles to tell Blender what it should use to deform: Vertex Groups and/or Envelopes. You may have noticed these options are repeated also in the Editbutton --> Armature panel, but as the tooltip says: these two are used when you use virtual modifier (the old way) to keep compatibility with old files.

Parenting the mesh to the "armature" will create an old-way link, still visible in the modifier stack, but not very useful. The first entry with the "make real" button is what appends if you do a CTRL-PKEY to "armature". You should not use that kind of connection when you see that. Press "make real" to get a working modifier.

157.2. The Old Way

This way is not recommended but can still be useful. When doing CTRL-PKEY to "armature", you will get a menu like this:



Figure 1019

- Don't Create Groups will just create a virtual modifier so you can deform the mesh (the "make real" button)
- Name Groups is almost useless now as blender will create a group for you when you do weight painting.
- Create From Closest Bones is a function to remember when you want to bake all your envelopes to vertex groups.

157.3. Tip: Bake envelope to vertex groups

The workflow is very simple. When you are done with the envelope's tweaking and you have gotten the best out of it, delete the Armature modifier and parent the mesh to the armature. To parent it, go to object mode, first select the mesh and then the armature, then press CTRL-PKEY. Select *Create From Closest Bones*.

Do ALT-PKEY and redo the Armature modifier. Now all the envelope influence are converted to Vertex Groups. This way you can further tweak influence zone using Weight paint. More info in the following pages.

158. Envelope

158.1. What is Envelope

Envelope is a new visual tool to help you rig your characters faster and easier. It can often save you a lot of time. Each bone has a special area around it, allowing you to tell Blender what part of the geometry will follow each bone. This zone is customizable so you can move, scale and blend them together.

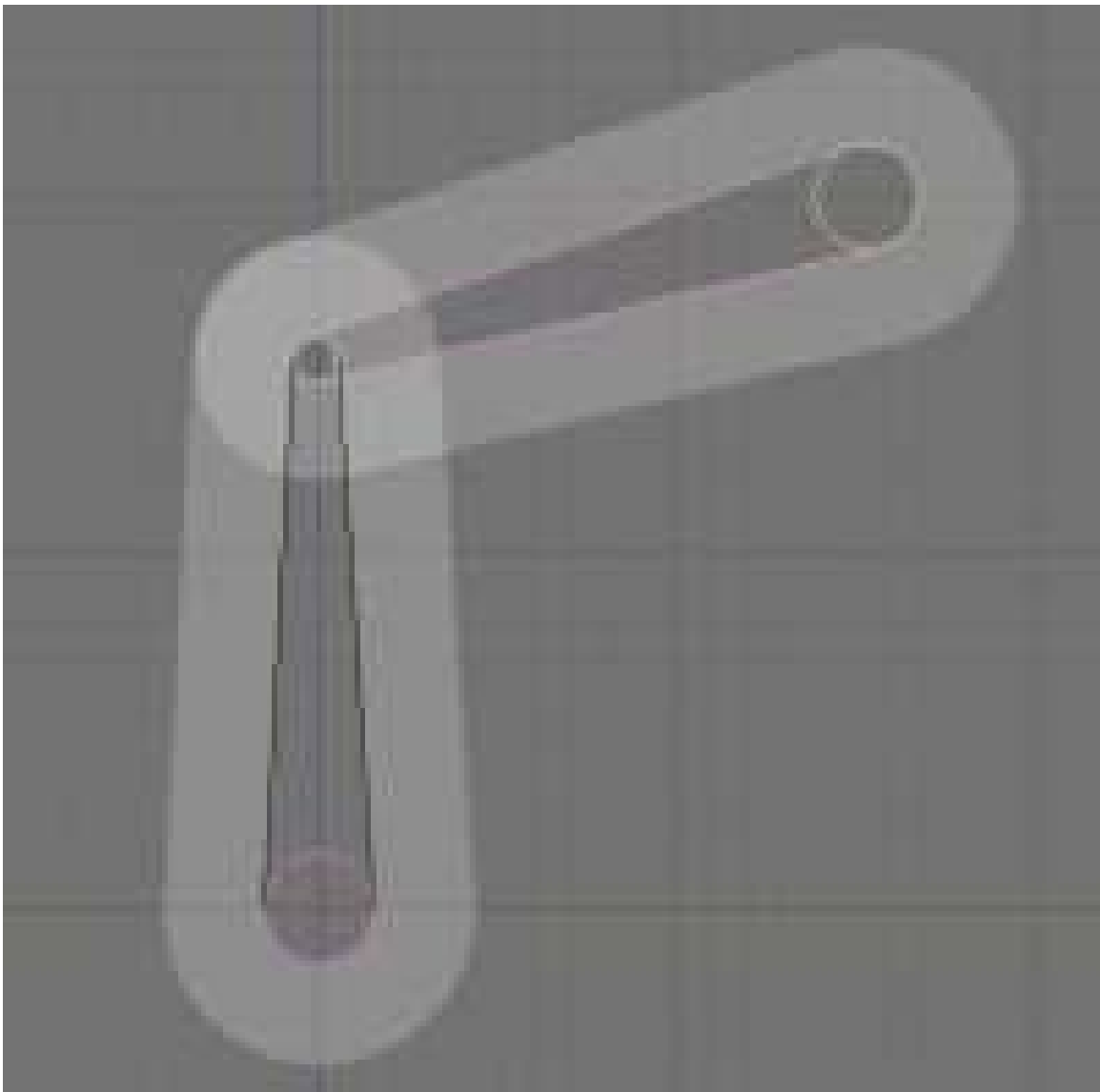


Figure 1020

158.2. Edit Envelope

You can edit this white zone in Editmode or posemode by going in Envelope display mode (Buttons Window → Armature → Display Options → Envelope), selecting bones and using SKEY or CTRL-ALT-SKEY.

In Editmode: you can select the Tip, the Body or the Root and scale using SKEY. This area in the middle will assign a weight of 1 to all vertices contained in here. All vertices with a weight of 1 will completely follow this bone. The white transparent area around the center is a zone of influence which loses power as you go away from the center. This area is scaled when selecting the body of a bone and doing CTRL-ALT-SKEY. *In Posemode:* You can only scale the zone of influence with ALT-SKEY when in Envelope display mode. It's real time, and lets you tweak the influence while you animate. So if you notice there is a vertex not following in the new pose you just did: Just select the bone it should follow, and scale the zone a bit until the vertex goes back with his friends. Example:

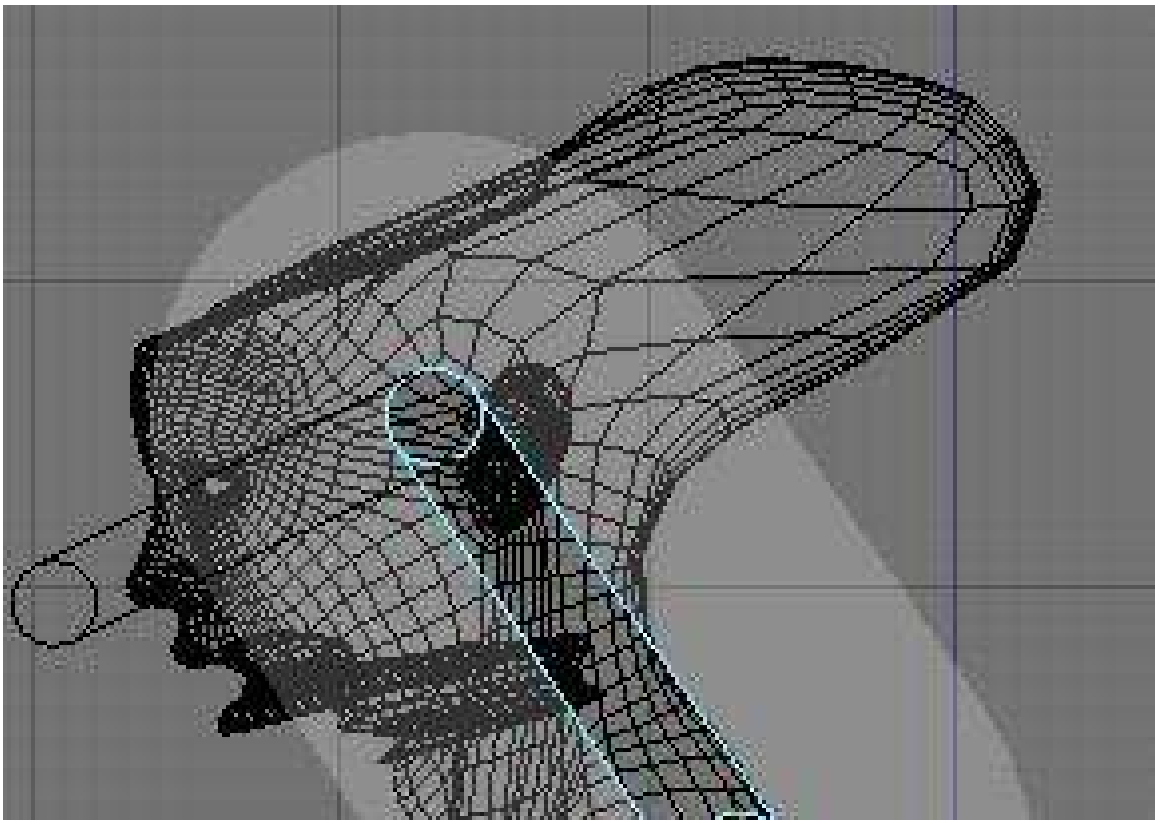


Figure 1021

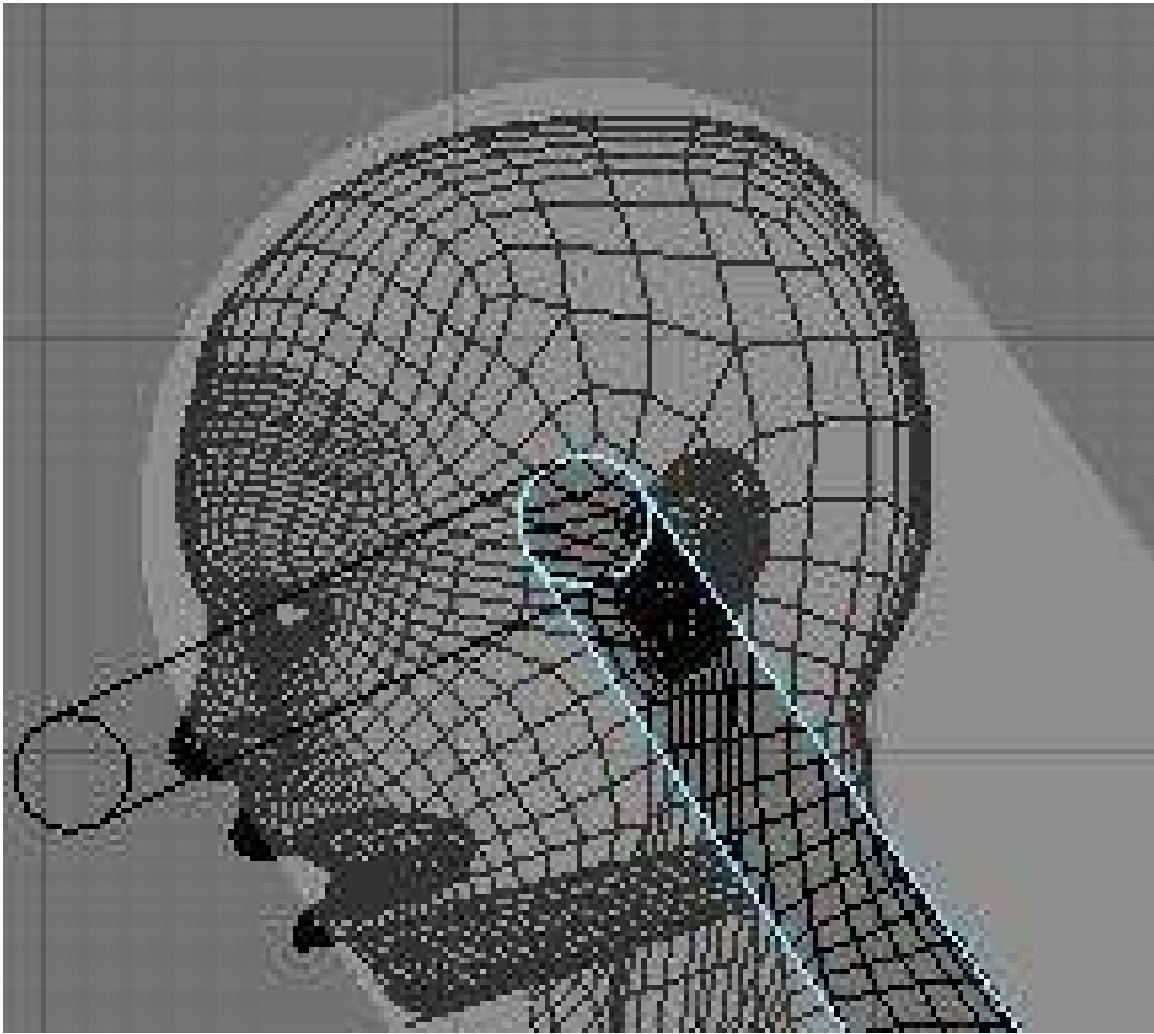


Figure 1022

158.3. Envelope Options

It's possible to enable/disable the use of Envelope in the Modifier stack using the "Envelope" toggle.

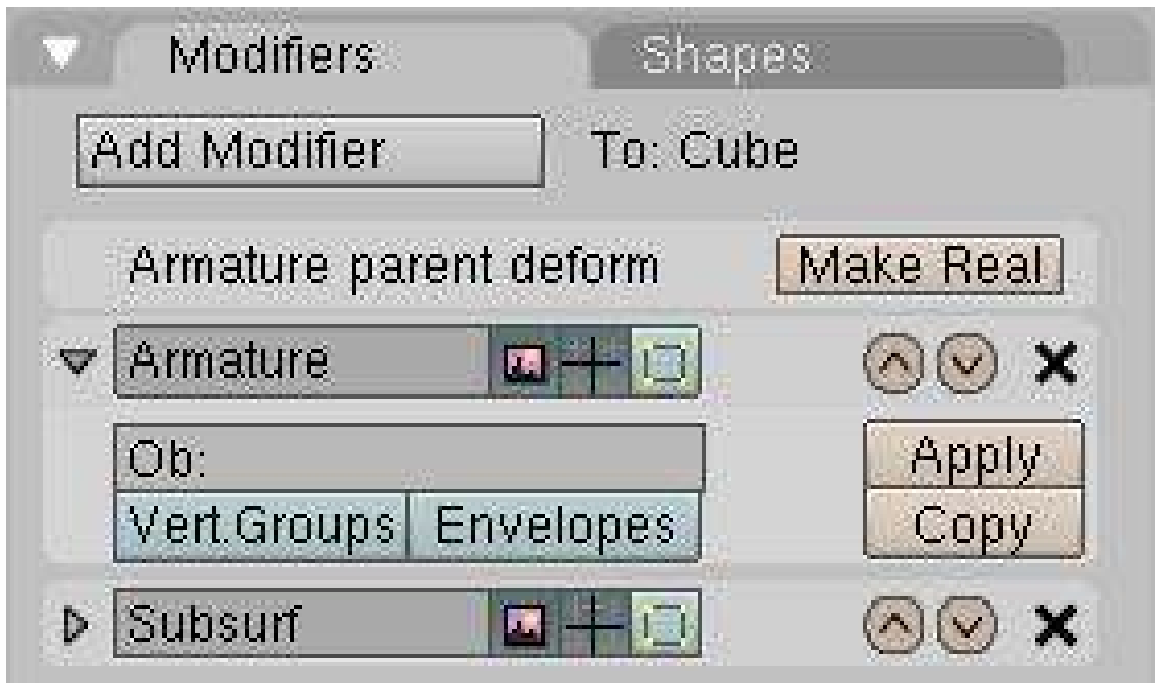


Figure 1023

There are also two important buttons in the Armature Bones panel: Deform and Mult.

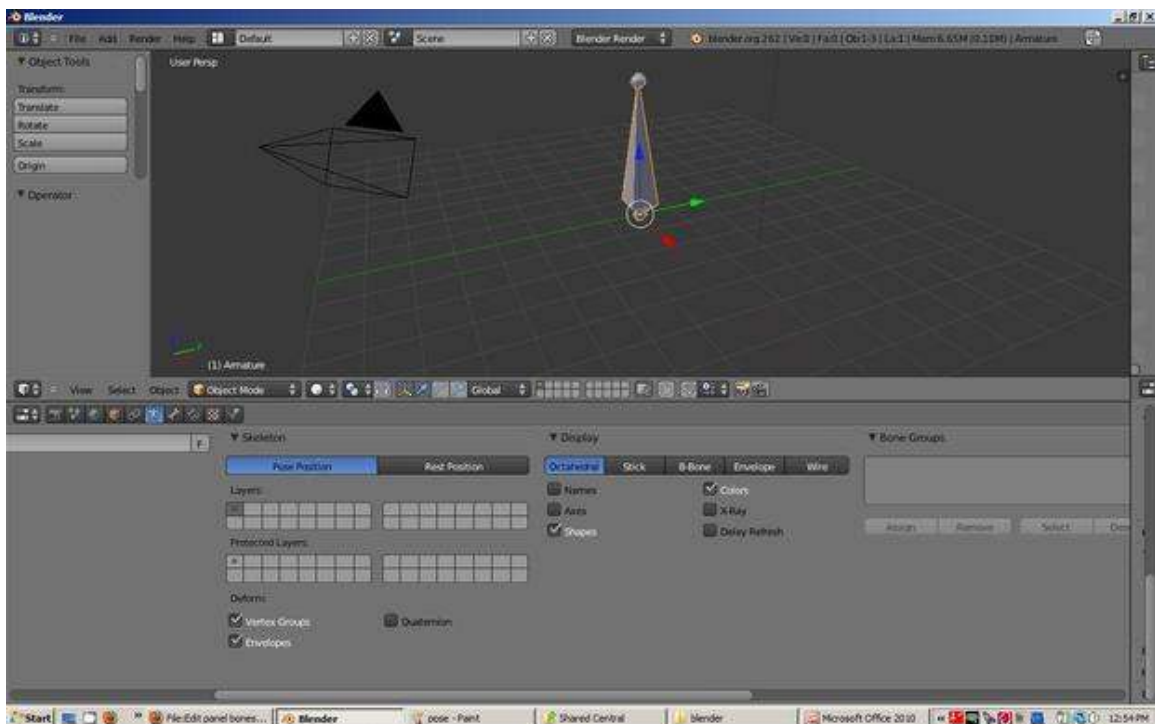


Figure 1024

Enabling the Deform button will tell Blender to deform geometry with this bone. It's useful because in a more complex rig not all the bones are there to deform, some bones are just there to move other bones.

The Mult option will tell Blender to multiply the weight it gets from envelope (let's say 0.7) with the weight you painted in weight paint (let's say 0.5). The result will be $0.5 * 0.7 = 0.35$ so in fact you just tweaked the envelope influence to 0.3 when it was at 0.7. If you don't want vertices to be part of the zone, you can always paint it with 0, as $0 * (\text{something})$ will always give 0. This way you can give custom shape to your envelope. More on weight paint on next page.

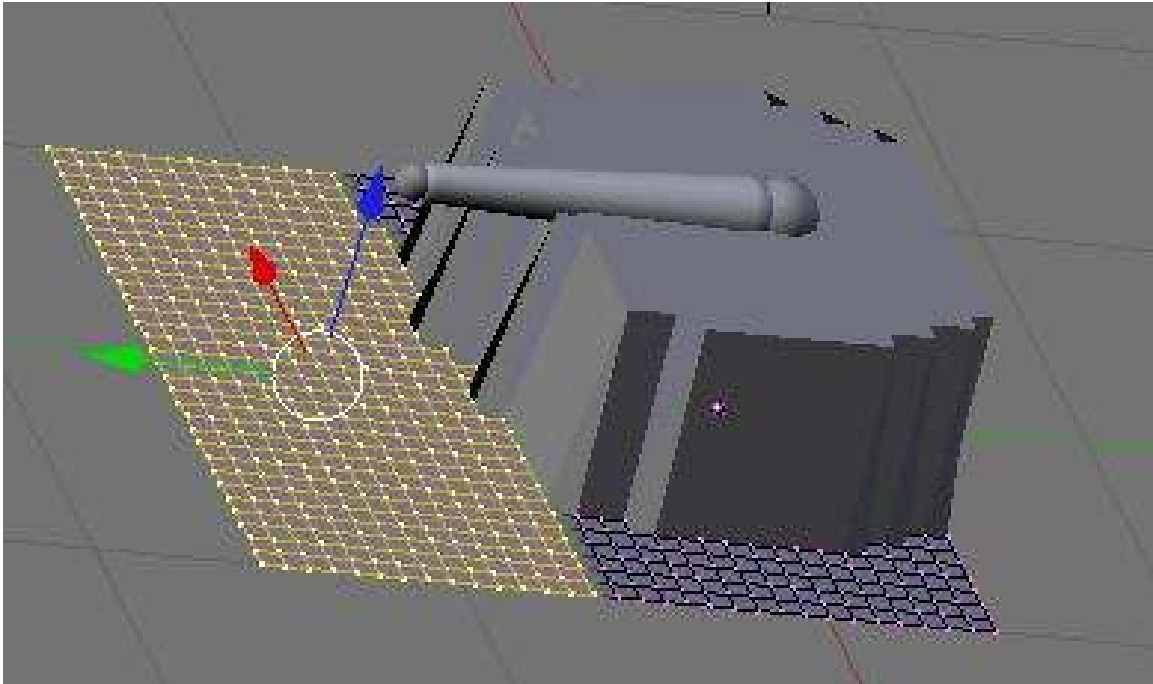


Figure 1025

In this example of a formerly flat surface you can see that all the selected vertices are not following the bone when it is moved upwards. This is because I painted a weight of 0 on them. In weight paint you'll see nothing. But just the fact that they are part of the group with a weight of 0 will make that possible. If Mult is off and you have both a vertex group and envelope, Blender will add value.

159. Vertex Groups and Weight Paint

159.1. What Are Vertex Groups?

Vertex groups are very useful. You can use a vertex group to:

- Group vertices together while you model (keep a selection and come back to it later).
- Define which vertices softbody simulation affects.
- Define which vertices emit particles.
- Define which part of a mesh will follow a specific bone.

Vertex groups are specific to the Mesh object and can be modified in Editmode.

If you have vertices assigned to multiple groups (for example, in a character you may have some vertices in the "upper arm" vertex group that are also in the "lower arm" vertex group), you can assign weights to those vertices to specify how much relative influence the different groups have. A weight can range from 0 to 1 and is assigned when you create the group. Let's take a peek at the GUI of vertex groups in the Editbutton(F9KEY):



Figure 1026

From top down:

- The dropdown menu lets you select an existing vertex group or rename the current one.
- The weight numfield lets you choose the weight value assigned when you add vertices.
- You can add a new group or delete the current one.
- Assign or remove selected vertices to/from current group.
- Select/deselect all vertices in current group.

159.2. Weight Paint

As mentioned above, you may often find that you have some vertices that are assigned to more than one vertex group. By assigning weights, you can specify the relative influence each of the vertex groups have. You have two options to assign weights: 1) manually selecting each vertex and typing in a weight value, or 2) use weight painting to - you guessed it - paint weights.

Weight painting lets you paint weight values on the mesh like you were painting on a wall with a can of spray paint. It is a Mode of the 3Dview and is accessible in the 3Dview's header in the dropdown menu with Objectmode, Editmode and such. A hotkey is also available: CTRL-TABKEY.

In Weightpaint Mode, the first thing you'll notice is the blue color of the mesh. Blender provides an easy way to quickly visualise the weight value of each vertex. This is the color spectrum used:

Figure 1027

When you are in Weightpaint mode you can paint all over the mesh as if it was a solid object on your desk. The paint only works on vertices so don't try to paint in the middle of an edge or a face, it will never work. To help you in your task there is a new panel in Editbutton:

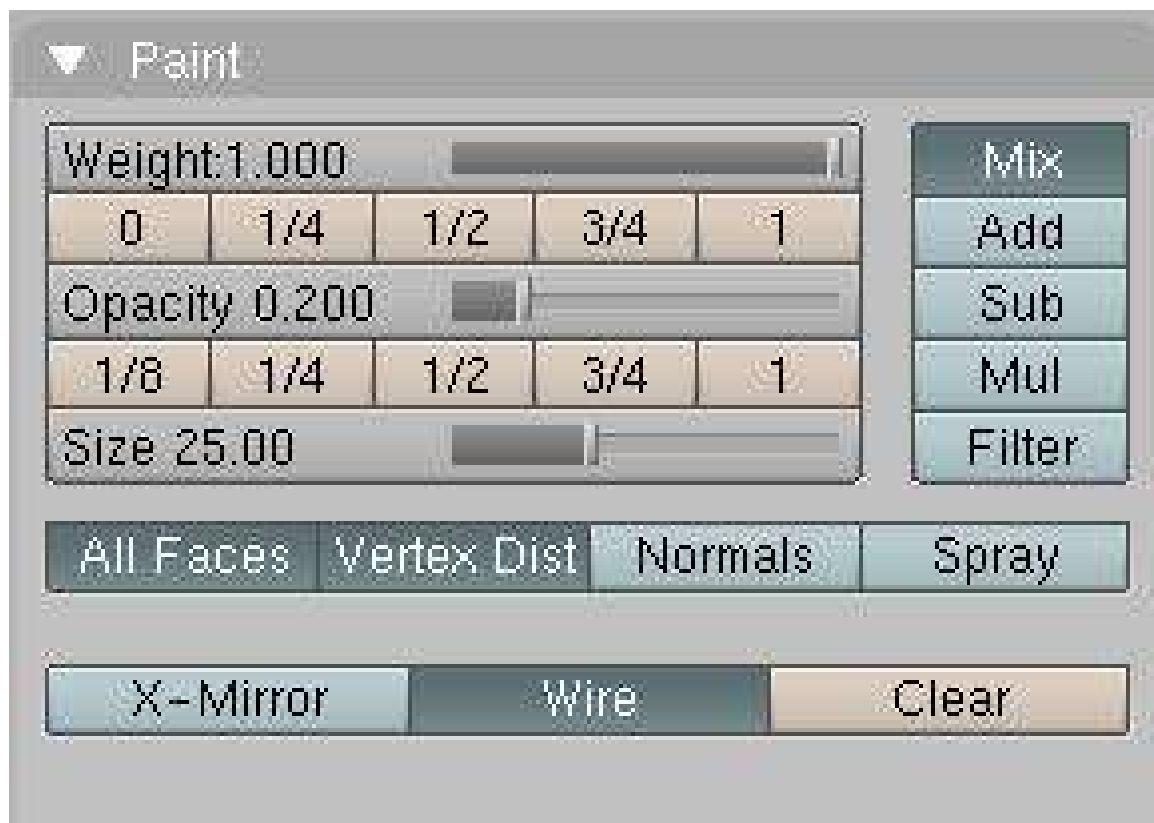


Figure 1028

- The weight slider is just the same thing as the weight numfield we saw earlier in the vertex groups button. It's just easier to work with. It's simply the weight you want to apply to the vertices. In painting terms, think of this as the color.
- The buttons from 0 to 1 are shortcuts for weight value, to speed up the workflow.
- The opacity slider (and shortcuts) tell Blender what is the percent of the weight value you want to apply in one shot. If you set opacity and weight to 1 the vertex will turn red instantly. In painting terms, think of this as the pressure.
- "All faces" tells Blender if you want to paint on all faces in the mesh or just the visible one.

- "Vertex Dist" tells blender to use vertex distance instead of faces. When active, the painting will only check if the vertex is in the brush, then apply a weight value. If it's off, all vertices part of the faces in the brush will receive weight value. Turning on Vertex Dist can give good results when you have a lot of polys in your mesh.
- "Normals" will apply vertex normals before painting. This means Blender will take consideration of the direction the vertex is pointing when painting: the more it's facing away from view, the less it will receive value.
- "Spray" really makes it like spraying paint. Without it, a single click will only paint one value. With Spray on, each time you move the mouse a paint shot will be added. To get a good effect, use a small opacity value so the weight will increase slowly.
- "X-mirror" will tell Blender to apply the weight paint on the other group if there is one. Like Hand.L --> Hand.R. If you paint the group hand.L and there is a hand.R the paint will be copied over. For this to work your groups must be created, the name of the groups have to follow name's convention (left right) and both side of the mesh need to be identical.
- "Wire toggle" toggles the visibility of wire while painting. Useful to find where the vertices are (activate the edit mode option "Draw all edges" to see even better).
- "Mix"/"Add"/"Sub"/"Mul"/"Filter" is how you want to apply the paint based on what is already there. Mixing will do a mean from brute weight value and current weight value, "Add"/"Sub" will directly add or subtract value, "Mul" will multiply (exponential painting) and "Filter" will paint based on alpha value.

159.3. Vertex Groups and Armatures

So what use are vertex groups in rigging? You can specify what vertices will move when a bone moves. When you want to paint a mesh for an armature, do the following:

- Make sure the Mesh has an Armature modifier.
- Select the armature and enable Armature Posemode (CTRL-TABKEY).
- Select the mesh and enter Weightpaint mode (CTRL-TABKEY). (**Question: if we're in Pose Mode for the Armature, how are we supposed to select the mesh?**)
- Select the bone you want to paint for with RMB.
- Paint the parts you want that bone to affect.

You'll notice that, if there is no group created when you first paint, Blender will create a group for you, and give it the same name as the selected bone. This is important, because when the "Vert. Groups" toggle is on in the Armature modifier, Blender will try to match bones with Vertex Groups based on the same names.

What happens when we try to blend groups together? See this simple example of 2 bones trying to bend a tube:

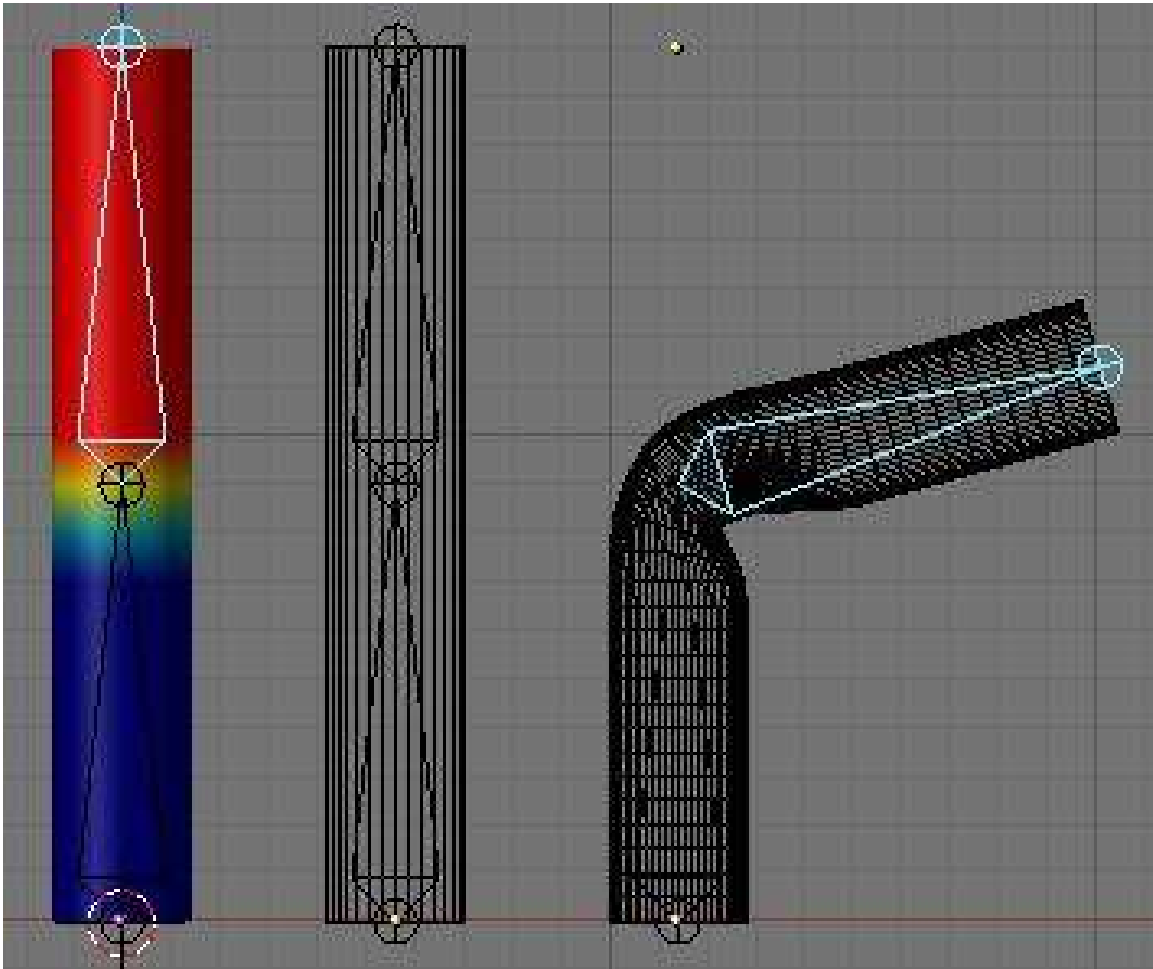


Figure 1029

The Groups are painted so the body of each bone is in red and the zone between the two bones are gradually going from 1 to 0. This will bend nicely. If, for a special reason, you want a side to react differently, you can always move the bone while painting and try the new modification you just did. By the way, having Subsurf on while painting can be very cpu expensive. It's a good idea to turn it off.

159.4. Using Weight Painting with Armatures

Armatures are used for many purposes, but one common use is to deform a mesh with an armature. This example will demonstrate how weight painting can improve armature-deformed meshes.

In this example, we have two objects; each has an armature modifier applied. The one on the left is going to be the "before" and the one on the right will be the "after".

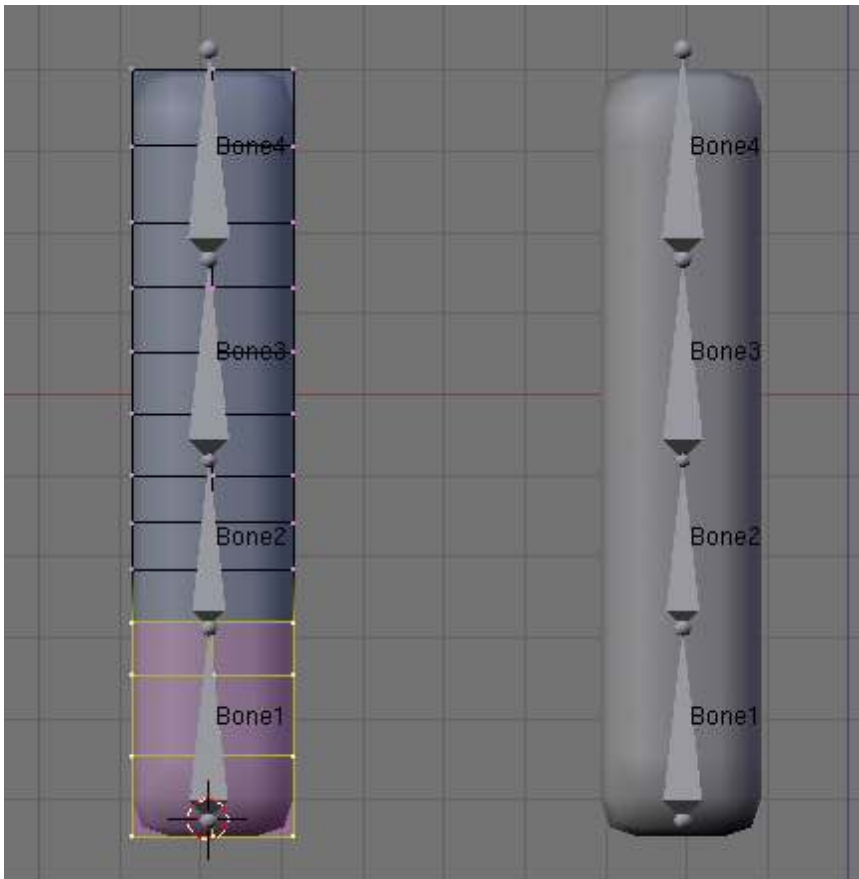


Figure 1030

For the object on the left, take a look at the vertex groups as initially assigned (from left to right: Bone1, Bone2, Bone3, and Bone4). These same vertex groups were assigned for the object on the right:

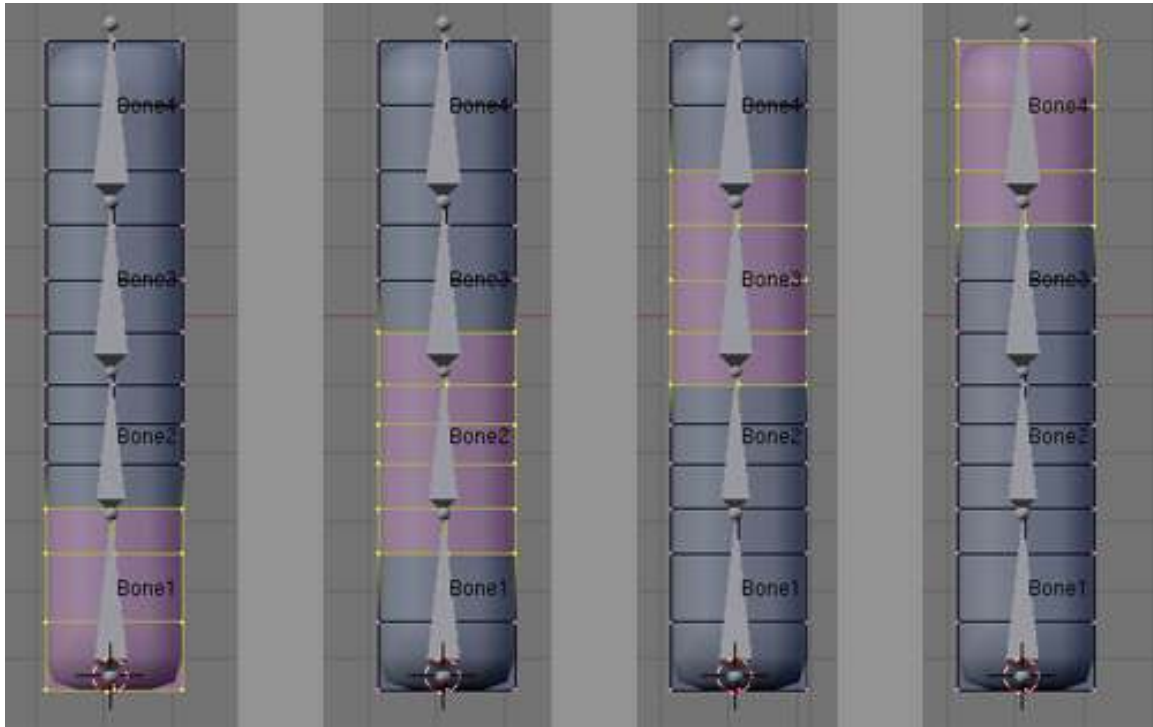


Figure 1031

Important: A bone in an armature will only act upon those vertices that are in a vertex group with exactly the same name as the bone.

- In Blender 2.37 and previous, this was the **ONLY** way to get a bone to deform a mesh.
- In Blender 2.40 and on, selecting the "Envelope" button in the armature modifier will allow bones to deform even if you haven't assigned any vertex groups yet.

If you enter Weight Paint mode (CTRL-TAB with object selected) right after assigning the vertex groups, you can see that the vertex groups as assigned all have a weight of 1.0:

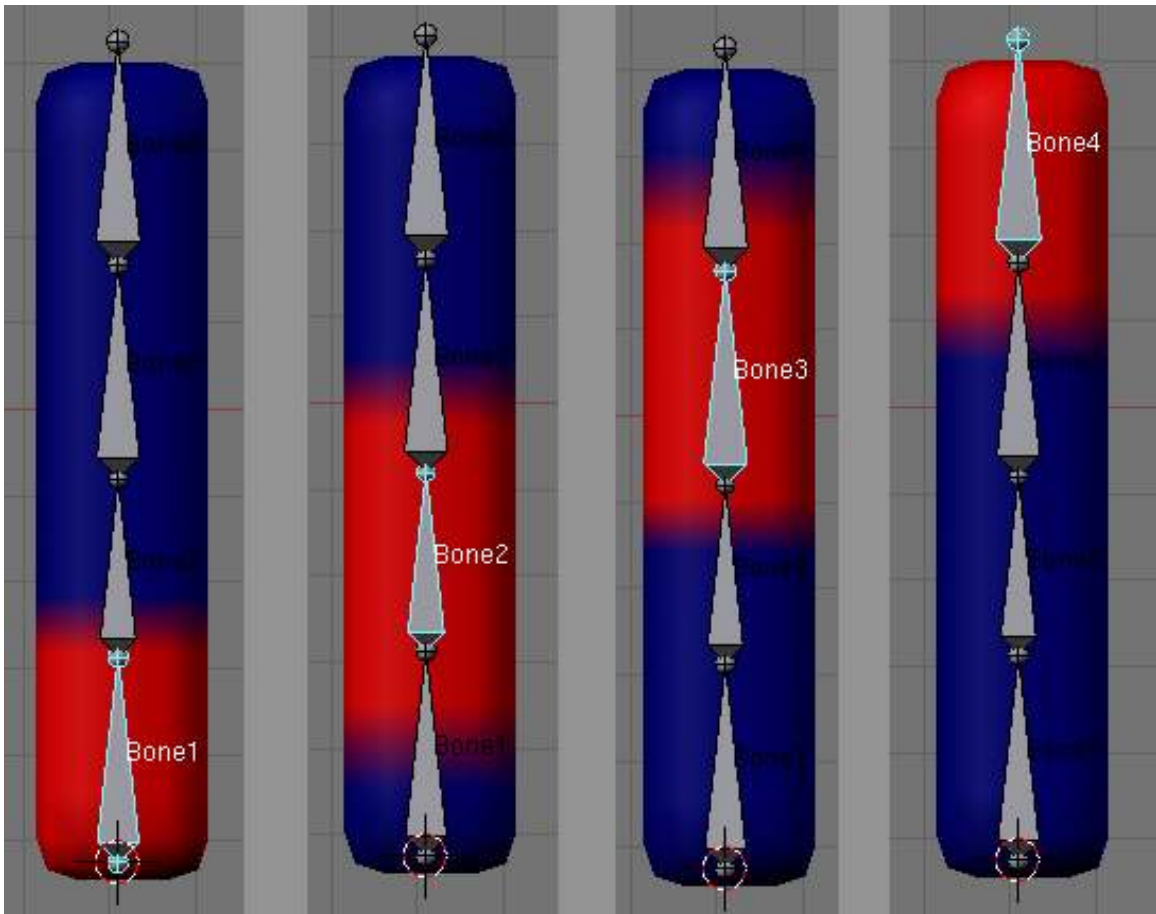


Figure 1032

OK: both objects have vertex groups assigned and they have armature modifiers. Let's grab a bone (select the Armature, CTRL-TAB to enter Pose Mode, select Bone4, GKEY to grab, and move) to deform the mesh. We haven't made the objects different from each other, so after moving their armatures in the same way . . . there's still no difference. That's good.

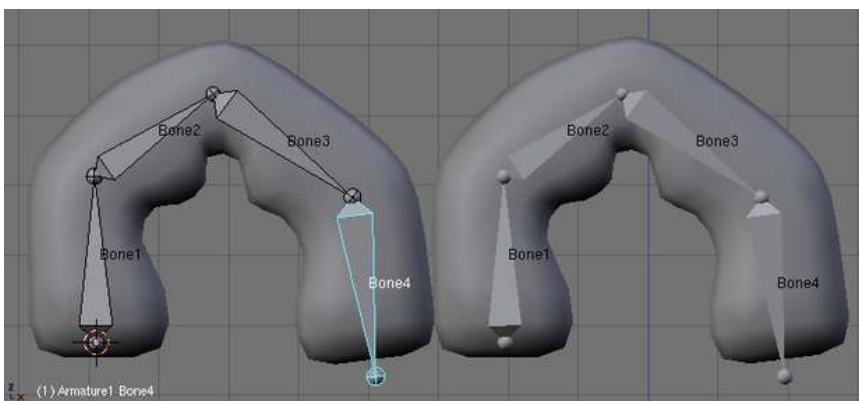


Figure 1033

160. Shape Key

160.1. Shape Key?

Shape keys are modifications of the original mesh that you can animate and mix with each other. Previously called Relative Vertex Keys (RVK), one of their major uses is to create facial expressions.



Figure 1034

The idea behind Shape keys is simple. The basic, undeformed mesh you start with is the "**Basis**" shape. You can add a variety of different versions of this shape and store each one as a Shape Key for use in an animation (other software sometimes refers to these as "morph targets"). You can't add or delete vertices as Shape Keys only store the positions of vertices and not the creation/deletion of them.

Ok, to start out, take a plane. I'll add a new shape to it and move a vertex away:

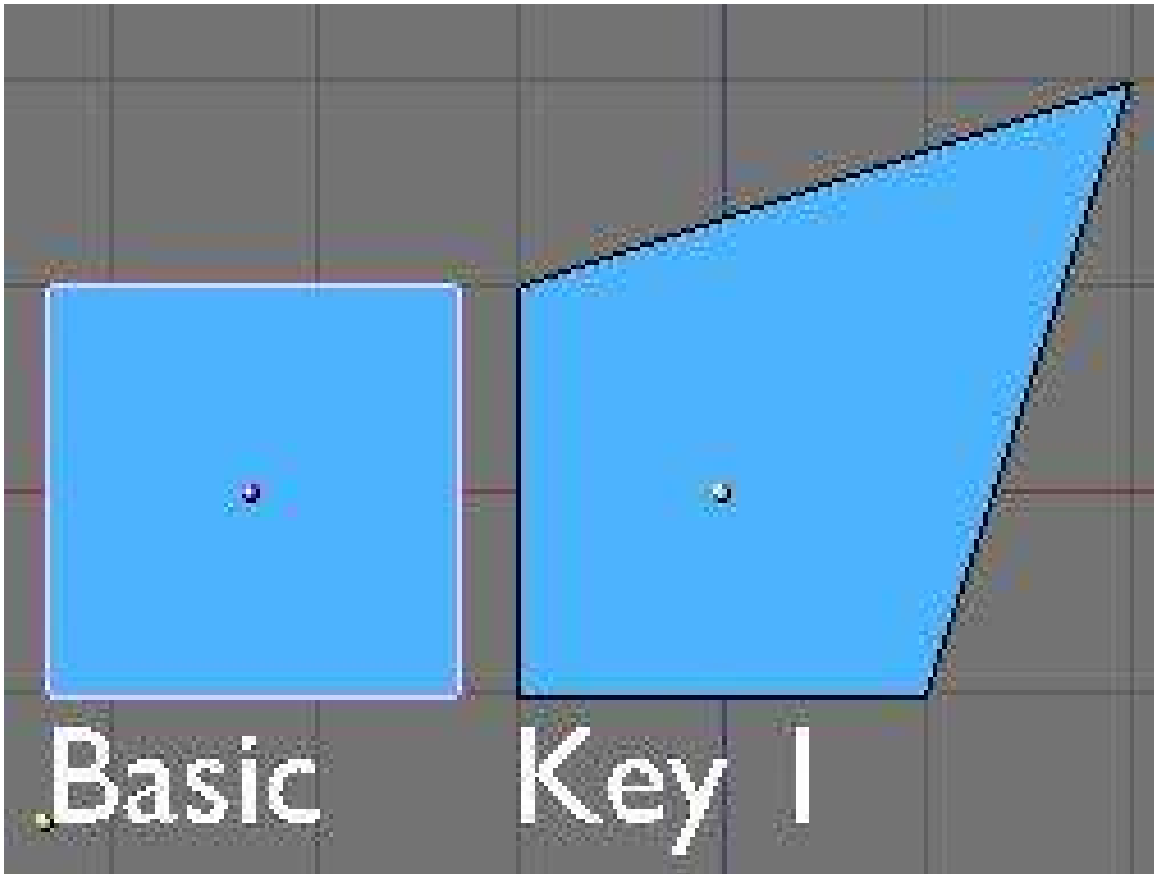
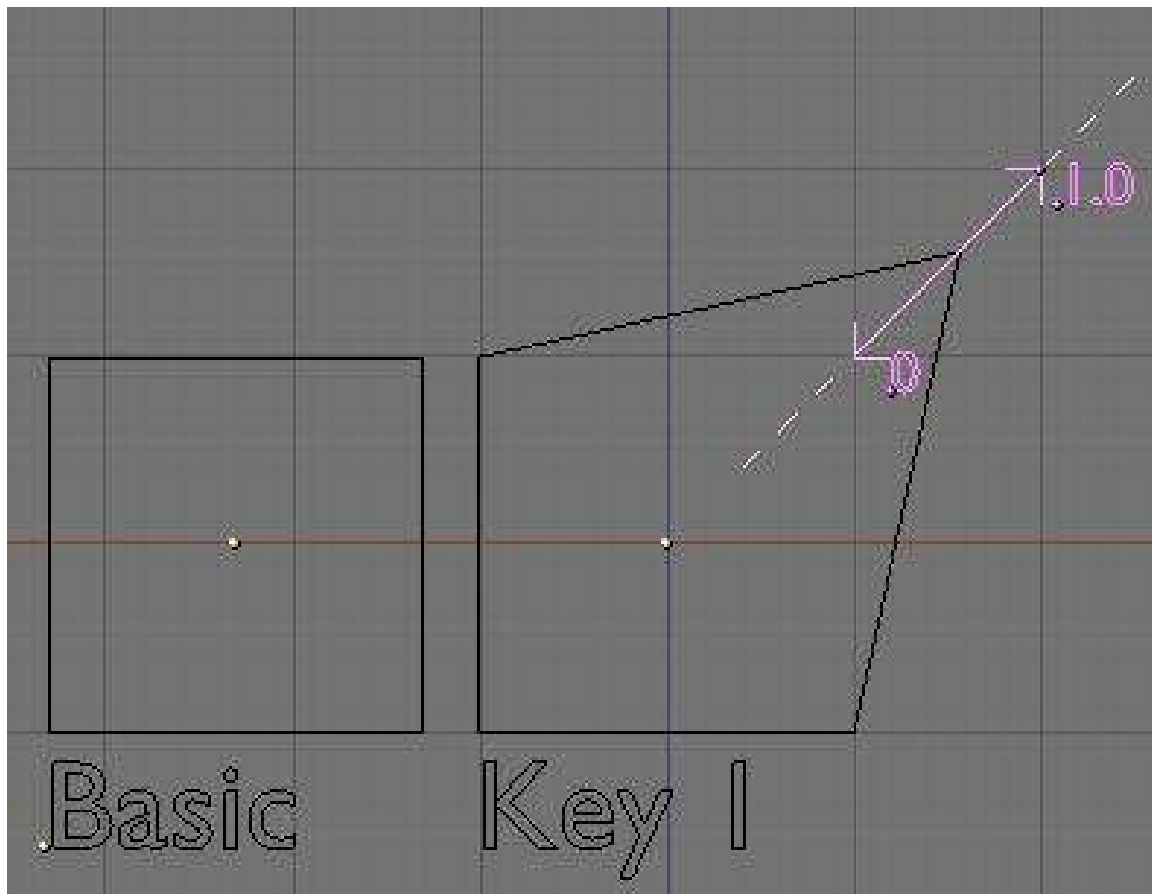


Figure 1035

When I play with the influence, this key has over the basic shape, the result will be as follows (0.5 in this example):

**Figure 1036**

If I play with the slider from 0 to 1, Blender will gradually apply the Key 1 shape with a constantly varying amount of deformation. So from 0 to 1, it will slide from the basis shape to the Key 1 shape. The movement of the vertices is linear from start position to end position and you can even set an influence greater than 1 or lower than 0 and Blender will extrapolate the movements of the vertices relative to the basis shape (marked as a dotted line above). This feature allows you to set general shapes such as a smile and then exaggerate the shape without needing to remake it.

160.2. The GUI

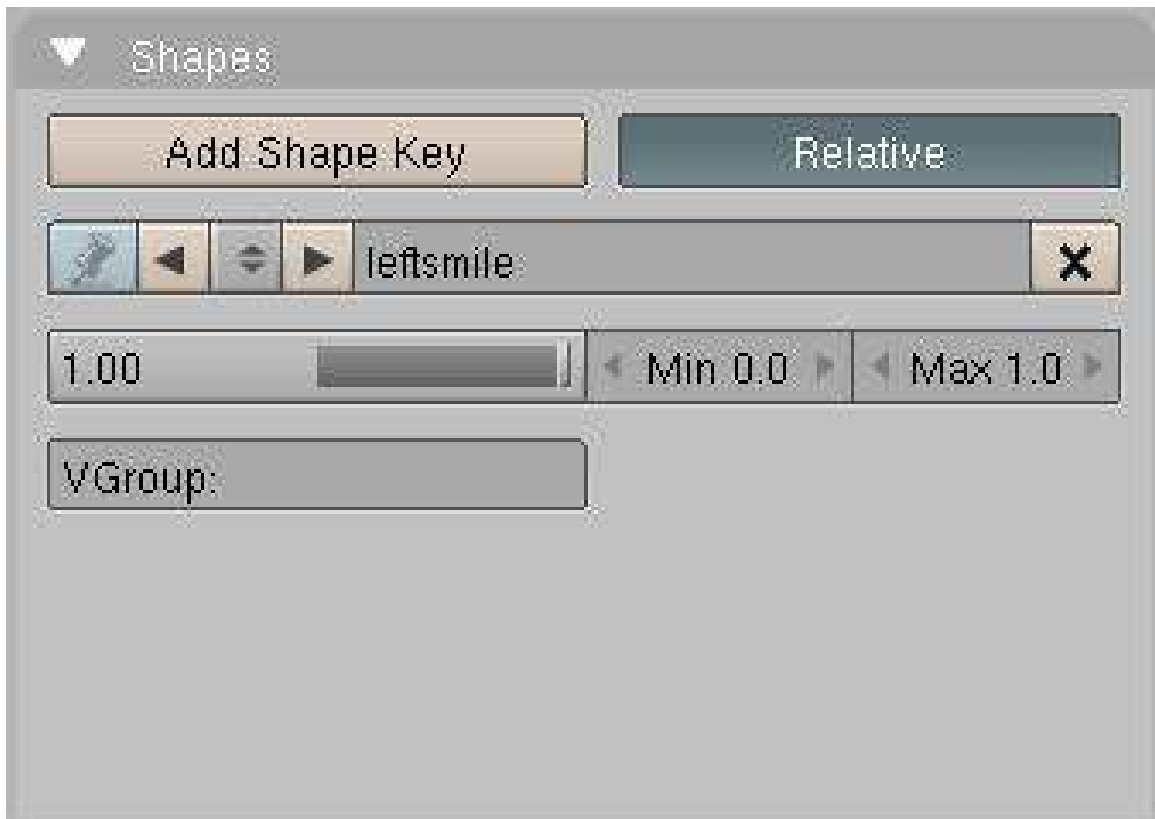


Figure 1037

160.3. Shape Keys step-by-step

Here's a hand-held walk-through of how shape keys are used.

Start with the default cube or a more interesting mesh of your choice.

1: Select your object in **Object mode**. Go to **F9 Editing** window. Find and select the "**Shapes**" panel. Press the "**Add Shape key**" button. This adds a key called "**Basis**" and this stores the "*basic*" mesh in its undeformed state. Make sure the "**Relative**" button is pressed (should be default).

2: Press the "**Add Shape key**" button again. It now says "**Key 1**" and you have a slider and some other settings. Go into **Edit Mode**, grab a vertex and move it. **Exit Edit Mode**. The mesh returns to normal but you've just added a real Shape key and Blender has stored it.

3: Repeat step 2 for as many different shapes as you like. You can rename the keys to whatever you want. Normally you create each new shape by first selecting the "Basis" key but if you select an existing key from the list and immediately press "Add Shape Key" then enter Edit Mode, the mesh will already be deformed. This is useful for making similar but unique keys.

160.3.1. Using Shape Keys

1: Starting on frame 1, select each key one by one from the pop-up list (*or go to the Action Window, press the sliders button and select the Keys from the list there*) and click on the slider and move the slider forward then back to zero. Sometimes just clicking on the slider at the zero point is enough to set the key.

2: Move forward ten frames, select Key 1 from the list and move the slider to 1.00. You'll see your object deform. Move more frames and slide another key. And so on and so on. You can move them forwards and/or backwards as you move through the frames and they'll just add together if they have to. Each time you move the slider, you set a keyframe for that Shape Key.

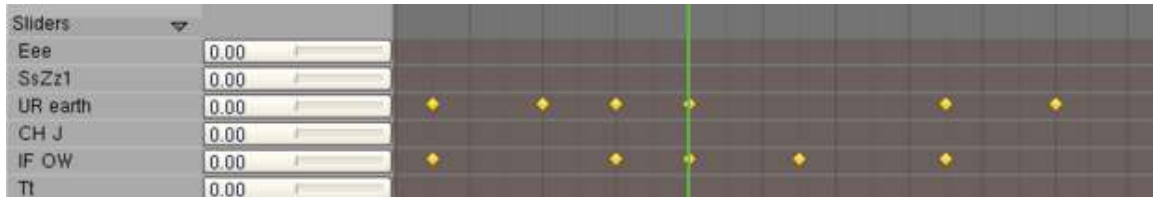


Figure 1038

Setting Shape Keys

160.3.2. IMPORTANT NOTE:

You should add shape keys to a finished mesh. Don't work on a mirrored mesh or a partially finished model. Adding geometry (vertices, faces, loops, joining etc...) can result in the loss of the shape keys or to unpredictable results. Not always, but probably when you least expect it. There are scripts available to make some of these things possible.

That's the basics. Note that shapes will constantly transform between keys so if you set a key at frame 1 to zero and at frame 10 you set the slider to 1 then at frame 5 the slider will be at 0.5 - which you may or may not want. If you want it to hold shape for a while before changing (*e.g. staying at zero until frame 7*), you'll need to set a key at the beginning and end of the time frame you want it to stay the same (*So you would set it a zero at the start, then zero again at frame 7 then to one at frame 10*). You can also edit the IPO curves (*use Shapes pop-up*) to really mess with the transformations.

161. Lip-Sync

161.1. Lip-Sync with Shape Keys

Here I will attempt to explain my recent dealings with using Blender Shape Keys to produce convincing lip-sync (Lip-synchronisation, i.e.: "speech") for simple, humanoid characters.

This is aimed at people with an understanding of Blender fundamentals like *vertex loops*, *face loops*, *sequencer* and of course, Blender's new *Shape Key* system. If these terms mean nothing to you, then you may well struggle to keep up. If you're familiar with them then I hope this tutorial will prove to be a breeze and your characters will be speaking so fluently you'll have trouble shutting them up!

Other Lip-sync tutorials, if you can find them, recommend using other software like Magpie, Papagayo and others but while I've no doubt they provide a valuable service and maybe make syncing easier, I will be using only Blender for this tutorial. I haven't tried using lip-sync software yet so I can't really say if it's easier or not.

You will find links to interesting audio files you can use for testing animation and lip-sync near the bottom of the page.

161.2. The hard work first

161.2.1. Setting up for Lip-Sync

First, set up your Blender screen so you have everything you need for this project. You'll need a front 3D Window, an Action Window, the Buttons Window (showing Editing - F9 - panels) and a Timeline Window. If you've got the room, a Side-view 3D Window would be handy too.

The basic sound units are called **phonemes** and the mouth shapes we use to represent these phonemes for lip-sync are called **visemes** (*or sometimes, phoneme shapes*) and there are many references for these around the web. Legendary animator Preston Blair created one of the most popular viseme sets¹, which is great for cartoon-style characters. Other visemes are aimed at more-realist, humanoid characters. The shapes you choose depend on the style of your model. (*I'll try to provide some useful links later*)

¹ http://www.garycmartin.com/mouth_shapes.html

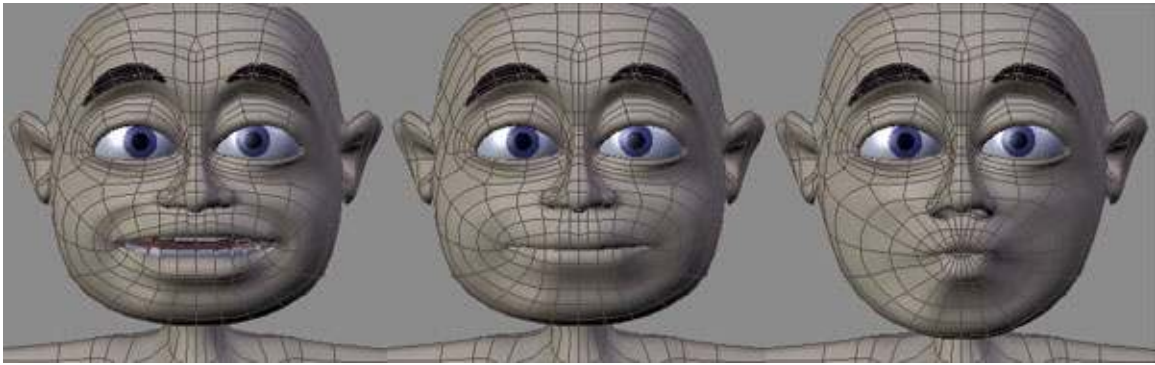


Figure 1039 Sample phoneme shapes

Sample viseme shapes.

The first and most difficult step in good lip-sync is making the shape keys for these visemes. How well these are made ultimately determines the success of the animation and it is worth spending time getting them right, although they can be modified and tweaked later. So choose your favourite set of visemes (*or even look in a mirror and use your own face as a reference*) and start setting your keys. A model with good **topology** - especially well formed **edge loops** around the mouth area - will be a big help here!

What on Earth are topology and edge loops?!?!

Topology refers to the way your 3D mesh defines the object, in this case your character's head. Edge loops refers to the flow of consecutive edges around the character's major features. Typically, good edge loops flow around the eyes and around the mouth in a somewhat deformed, "circular" manner. Selecting and deforming a single loop of edges and vertices surrounding a facial feature is much quicker than having to individually select a lot of edges that don't flow naturally around that feature. Similarly, deforming a single loop or collection of related, nested loops is much easier and quicker than trying to deform a seemingly random set of edges. You can easily see these loops in the character screenshots above. The series of edges surrounding the mouth simply stretch from wide ellipses to almost circular to create a useful variety of mouth shapes. The faces defined by paired edge loops are referred to as **face loops**. The close-up image below shows selected face loops.

There are many tutorials on the web about these topics so if you need more information before proceeding then a quick search is probably a good idea.

Setting the basic viseme shape keys

First, select your undeformed mesh and create your basis key. Go to **Editing** (F9) window and go to the "**Shapes**" panel. Press the "**Add Shape Key**" button. Enter and exit edit mode to set the basis key. Then create your first key **Key 1** and name it "M". Enter edit mode and if your character's mouth isn't already closed (*some people make them that way*) then close it by carefully selecting the faces and loops around the mouth. You will usually use Size-Z plus a bit of grabbing and shifting to achieve this. Don't forget to include the faces on the inside of the lips or the deformation will be unpleasant. When you're happy with the shape, exit Edit mode and there you have it. You character

can now say "Mmmmmmm" whenever you like. Test it by moving the Shape Key slider back and forth but make sure to leave it at zero before making more keys. (*If you made your character with a closed mouth then you can just add the new "M" key then enter and exit edit mode to set it.*)

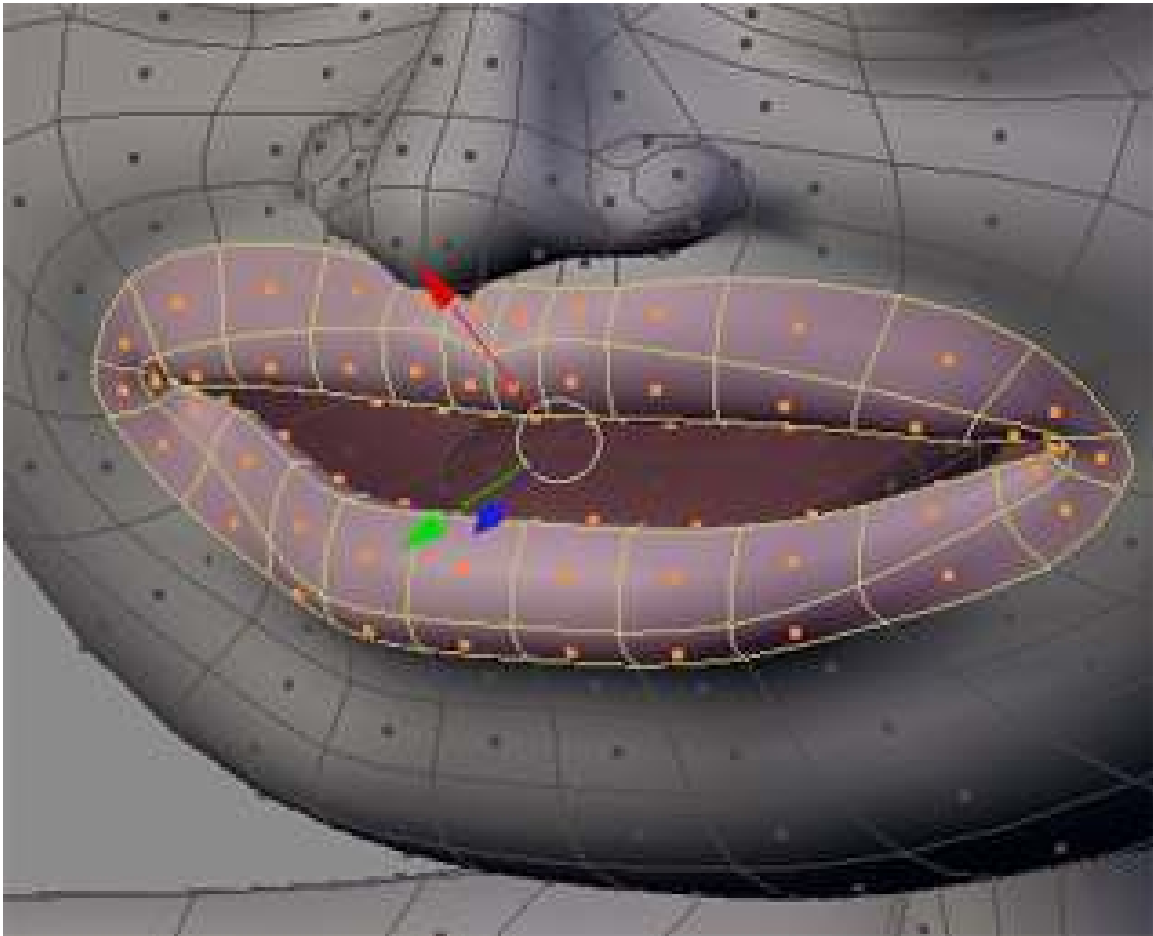


Figure 1040 Selecting face loops

Selecting face loops

For most new keys, you will select the basis key first then press "**Add Shape Key**" then make the required shape from scratch in edit mode. However, some mouth shapes are very similar, like "OH and OOO" or "EE and SS" and it would be easier if you could start with something close to what you want rather than shifting everything from scratch every time. Luckily, Blender allows you to do just this. Once you've made your "EE" key, for example, you can select it and immediately press "**Add Shape Key**" then enter edit mode and the mouth will already be deformed and you only need to make subtle changes to it to make your "S" shape.

Remember that the keys you make are for sounds, not letters.

In general, you'll need shape keys for the following sounds

- **M, B, P**
- **EEE** (*long "E" sound as in "Sheep"*)
- **Err** (*As in "Earth", "Fur", Long-er" - also covers phonemes like "H" in "Hello"*)
- **Eye, Ay** (*As in "Fly" and "Stay"*)

- **i** (*Short "I" as in "it", "Fit"*)
- **Oh** (*As in "Slow"*)
- **OOO, W** (*As in "Moo" and "Went"*)
- **Y, Ch, J** (*"You, Chew, Jalopy"*)
- **F, V**

These can often be used in combination with each other and with jaw bone rotations and tongue actions to also produce other sounds like "S", "R", "T", "Th" and a short "O" as in "Hot" - or these can also be specifically made with their own shapes. This decision depends largely on your character. Start with the essentials and make others if you need them.

NOTE: I use one jawbone in my current character and this is also used to control the mouth shapes. It doesn't drive the shapes but it moves the bottom teeth and tongue (*which can also be controlled separately*) and the faces that make up the chin part of the character mesh. For some visemes, I move the jawbone into a logical position before adding the shape key. For example. I open the jaw for the "OH" key and close it for the "M" key. Later, when animating, the jawbone is animated along with the Shapes for a very convincing result.

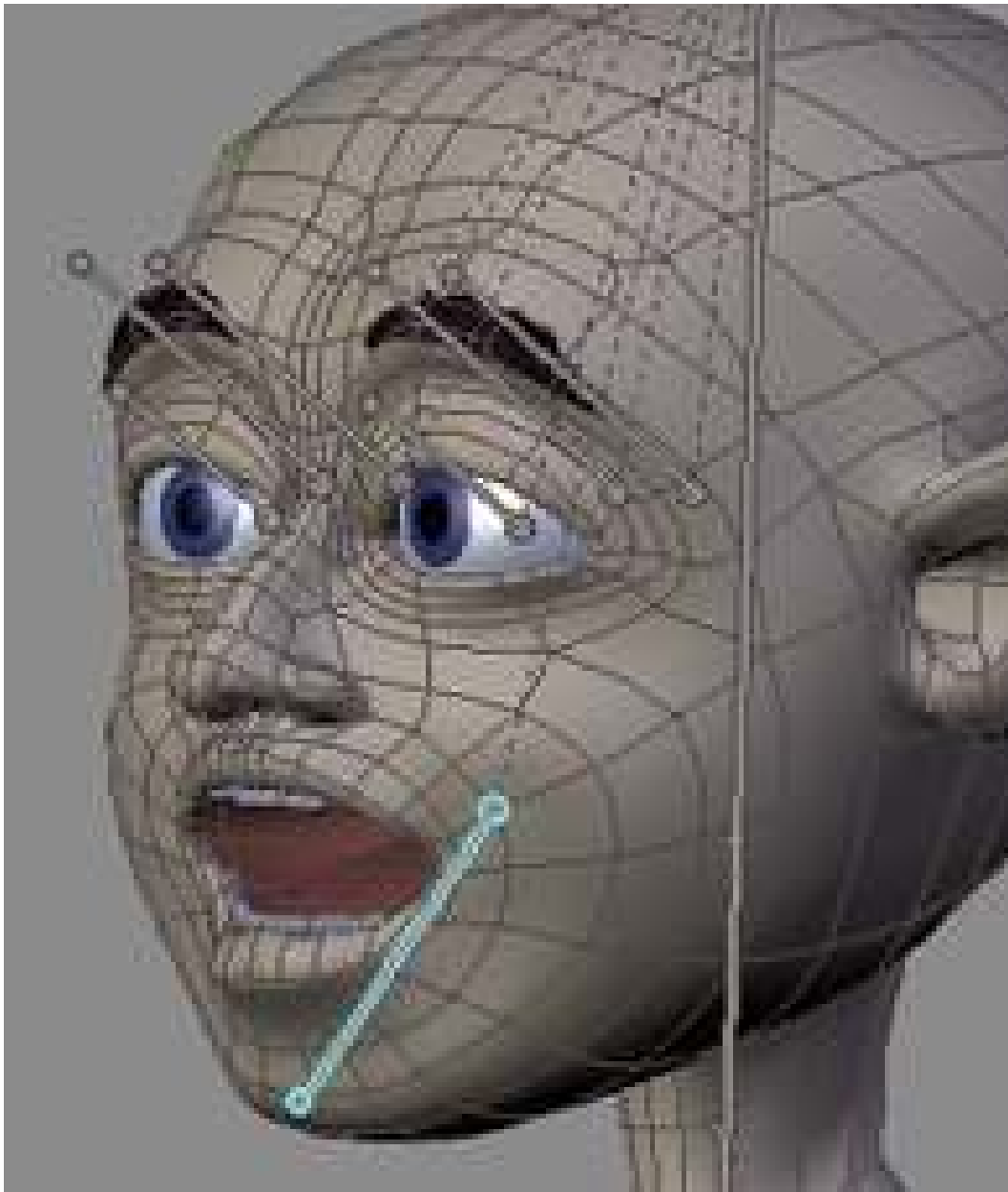


Figure 1041 Using jawbone with shapes

Using jawbone with shapes

Load the audio

Once all your viseme shapes are set the time comes to get your character to speak. (*Normally you would animate the body first and leave the lip-sync till near the end.*)

If you haven't already done so, load your audio file into a Blender **Video Sequencer Window** and position it where you want it. Currently, Blender only supports 16 bit wav format audio files so you

may need some editing or conversion software to process the file if it isn't in this format. "Audacity" is a good, open-source (free) editor that will suit this purpose and much more.



Figure 1042 Blender Soundblock Panel

Blender Video Sequence Editor Window showing loaded Audio Strip

Go to the Scene Window (F10) and press the "Sound block buttons" button --(in the three buttons near the frame counter. It looks like an audio wave)--. Press the "Sync" and "Scrub" buttons. The "Sync" button makes the playback in the 3D window closely match the audio when you press **Alt-A** (it does this by dropping image frames if necessary and it generally provides only a rough guide of how things match-up). The "Scrub" button is important for lip-sync as it means that whenever you change frames, Blender plays the audio at that point (Currently in some Blender builds you have to press Alt-A at least once to get this feature to start working).



Figure 1043 Blender Soundblock Panel

Blender Soundblock Panel

Select your character and your Shape Keys should be listed in the **Action Window**, in the order in which you made them (*I don't think they can be shuffled*). You will see a small triangle button

labelled "**Sliders**" at the top of the list - press it to show the sliders for each shape. If you drag the mouse back and forth in this window, you should hear the audio play as you cross frames. This is how you will identify the main viseme/phoneme key frames.

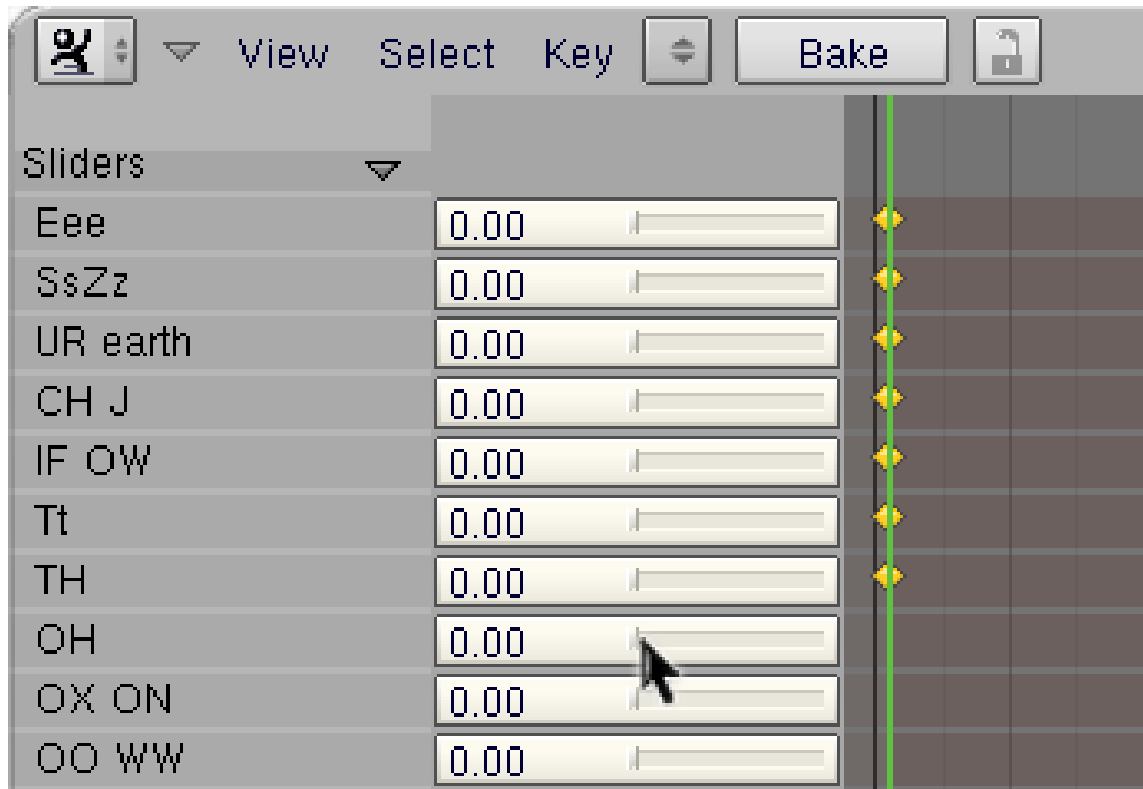


Figure 1044 Shape Key Sliders

Shape Key Sliders

161.3. Getting down and dirty

Before you begin your lip-sync, you obviously need to know what your character says. It might be worthwhile writing it down and even speaking it over and over, in time with the audio, to get a feeling for the sounds you'll be dealing with. Some sounds are what I'll call "**key sounds**" and others are almost dead, non-descript "in-between" sounds that fill in the gaps between the key sounds. Obvious key sounds are those where the lips close and those where they are tightly pursed or wide and round, other key sounds can differ with every piece of audio. Don't make assumptions about the shapes you'll use based on the words you know are there. What you are animating are the actual sounds - not letters or words (*Keith Lango has much to say about this on his website and I recommend reading it*)

161.3.1. The Timeline Window

Go to the **Timeline Window**. This window seems to have been largely overlooked in previous documentation yet it provides the basics for timed animation plus a few other goodies to make your

animation life a little easier. Here you can turn auto-keying on and off at will (*you'd normally go to the hidden **User Preferences** window which is a pain and easy to forget*), navigate frames, play and pause the animation in the 3D window, turn Audio Sync on and off, set the start and end frames for the animation and **set frame markers**. This last feature is the key to our project.

In recent Blender releases, the audio "Scrub" feature works in most windows. As you scrub through the audio, listen for where the key sounds occur. As you hit a key sound, scrub back and forth to find where that sound commences. In the Timeline Window, press "M" to set a frame marker at this frame. The marker can be labelled by pressing CTRL-M in the Timeline Window while the marker is selected (*yellow*). Enter a sensible name for the marker that indicates what the marker is for (*like the phoneme sound and/or which word it starts or ends*). Markers are selectable and moveable and can be deleted just like other Blender items.

NOTE: Depending on the speed of your system, you may find you get more pleasing audio scrubbing and better 3D window playback if you turn off subsurf for your model and hide any unimportant parts of the scene on different layers. The fewer things Blender has to calculate as you scrub or play-back, the faster it can draw the frames to maintain sync with the audio.



Figure 1045

Turn off these buttons to disable subsurf setting in 3D window



Figure 1046 Timeline Window

Blender Timeline Window

Scrub through the whole audio - or sections of it in a long piece - marking and labelling all the key sounds.

161.3.2. Setting the keys

Now you have everything you need for your first lip-sync pass. Start at the first frame and click once on all your **Shape Key sliders** in the Action Window to **set them all to zero**. Move through the frames from start to finish setting shape keys where you marked the key sounds in the Timeline Window. If your character has a jaw bone and tongue bone(s), you will need to set these where required as you go.

Trouble in paradise? A quick lesson in handling Shape Keys

If you haven't set shape keys before you might notice one interesting dilemma - they have great memories! Once you set a slider to any value, it stays at that value until you set another value somewhere. The shapes change in a linear fashion between keys. At first, this appears to be a problem if you want to key "MOO" because after you set the "M" key slider to 1 (one), it will stay there, making it impossible to get your character to say "OO" while his lips are trying to stay shut. So, you have to set the "M" slider on the "M" sound, then as the audio goes into the "OO" sound, you must set the "M" slider back to zero and then set the "OO" slider to its full value.

This introduces another problem. After you set the "OO" key, your "M" sound is messed up because the mouth is now also being affected by the "OO" shape that follows it. So, you must make sure the "OO" sound is set to zero when you want the lips closed in the "M" position.

In general, you'll find yourself setting each shape 3 times

- once to determine where you want the mouth to begin moving to this shape (*slider set to zero*)
- once to set the slider at the desired level for this phoneme
- and once more to end this shape and move into the next one (*slider set to zero*)

The same principle applies to the jaw bone and tongue - 3 keys to each move.

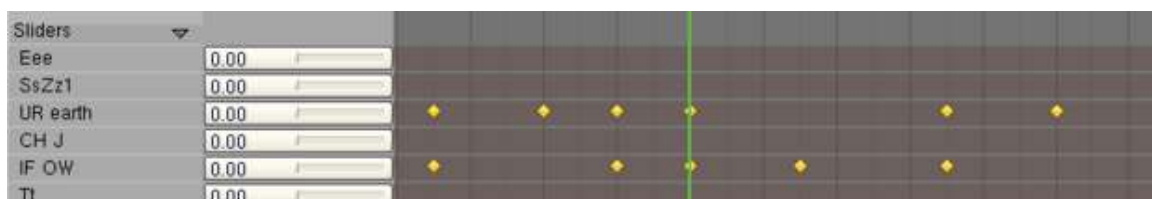


Figure 1047 Varying shape key settings

Setting multiple shape keys

As you get more comfortable with this procedure, you'll find you can leave some shapes set longer or adjust them to different levels as they can provide some interesting mouth shapes when combined with other shapes.

161.3.3. Setting the in-between sounds

Once the key sounds are properly set, you should be able to scrub slowly through the Action Window and watch your character speak in time with the audio. It won't be perfect but it's a start. To fix his speech impediment, you now have to fill in the sounds between the key sounds. Mostly these will be

dull vowel sounds ("err, uh, ah") and silence. These shapes are set in exactly the same way but here you'll have to really watch the 3D view as you set the sliders. Try combinations of sliders like "EE" and "OH" to get the perfect shape for each individual sound. You have to be guided by your character. Does he look like he's saying the sound you're hearing? (*Remember that exactly what's being said is not important - it's only the sound that matters*). Test each sound as you set it by scrubbing a few frames over and over and watching your character mouth the sounds.

All that's left, hopefully, is some polishing and tweaking. If it's not perfect then don't despair. Like other areas of animation, it can take a while to get a feeling for lip-sync and as the tools and workflow become more familiar you can pay more attention to the important work.

161.4. Putting it all together

Once you're reasonably happy, it's time to combine the audio and video and watch the result. Since Blender can't do muxing (combined audio/video) you'll need to composite it with the editing software of your choice (OSX users can use recent versions of iMovie, virtualDub is often recommended for Windows users and Avidemux2 is often recommended for Linux users.).

What Blender can do is provide a **fully synced** version of the audio file the same length as the animation - even adding silence at the start and end if need be to maintain the synchronisation. To make this synced audio file, go back to the "Sound block buttons" panel and press "**MIXDOWN**". This saves a .wav file using the filename and location you entered in your output box (*you did didn't you?*) plus a frame count (*something like 'speech.avi0001_0400.wav'*). Then save your animation by pressing the "**ANIM**" button.

Combine the audio and video in your video editor and export a muxed file. You may find when you play it back that the mouth seems to be just slightly out of sync. This is for two reasons: sound travels much slower than light, so we see the lips move before the sound reaches our ears, and the lag is more the further away the person is (about 3 milliseconds of delay for every metre), and the more exaggerated their expressions as they shout. The second reason is the way the brain processes faces and expressions and mixes it with sounds heard to comprehend speech. This comprehension phenomenon is barely understood and is a common challenge in animation, and some physiologists think our brains read lips and facial expression as a way of setting up to understand the context of sounds received and comprehending the meaning behind language. To solve it, you can go back into Blender, select the audio in the Sequencer Window and move it one or two frames later, then press "MIXDOWN" again to create a new .wav file with a split second of silence at the start. Remix this with your video and watch the results.

From here on its all a matter of testing and tweaking until you're happy!

161.5. Using Blender to render Audio AND Video TOGETHER.

Ever since Blender 2.3, it has been possible to render video with the audio attached. Once you have your animation done, and are ready to render: In the buttons window, click scene (F10). In the format tab, choose FFMpeg. There will now be three tabs. Click on Audio. Click Multiplex, and choose your audio codec. Click Video. Choose your video codec (including AVI). Now, in another section

of the window, there will be an Anim tab. Click Do Sequence. And click ANIM. When you watch the video, you'll notice that it has sound.

Note: The mac version of blender doesn't have the FFMpeg codec in blender by default.

161.6. Audio Files

You can find some interesting audio files selected for animators at <http://www.animationmeat.com/practice/practice.html>²animationmeat. These files come complete with a pre-marked phoneme sheet.

161.7. P.S.

One final note. Watch how the pros do this. When actors are doing voice overs for major releases their actions are recorded and even integrated into the final animation. If you have a digital camera, you may also try recording your own mouth performing your dialogue and approximating its positions to your animation. This can give you a great visual reference, possibly frames for frames if your frame rates match, and save you time.

² <http://www.animationmeat.com/practice/practice.html>

162. Constraints

162.1. The Constraint

A constraint is what makes everything easier, magic, automatic, customised (add more words here) in a rig. It tells a bone or an object to do something special based on the position of another object, and the position of the constrained object itself. There are many constraint types for you to play with. Most will work everywhere but, the IK solver will only be available in the Armature Editmode or Posemode.

There are no strict rules to follow when using constraints. As long as they save you time and make everything work by itself. A constraint should never be "time-consuming" or difficult to use. Think about the animator who is going to work with this rig (it could be you!). So, do everything in a smart way.

It's possible to copy constraints from one object/bone to a bunch of objects/bones. A useful thing to know when doing a repetitive task like rigging all the fingers of a hand. Just select all bones/objects that you want to give a copy of the constraint, and then select the bone/object containing the constraints. Press **CTRL-C** in 3DView, and select Object Constraints from the popup menu. The idea behind this is to copy the constraints of the active object to the selection.

When working on an armature in Posemode, the bones will change color if they contain a constraint. Green for almost all, except for the IK constraint, which turns the bone Yellow.

162.2. The Constraint Panel

You can add a Constraint to an object or a bone by going in Object button window(**F7**) for objects and bones. Look for a Constraint panel like this (note, it's usually empty):

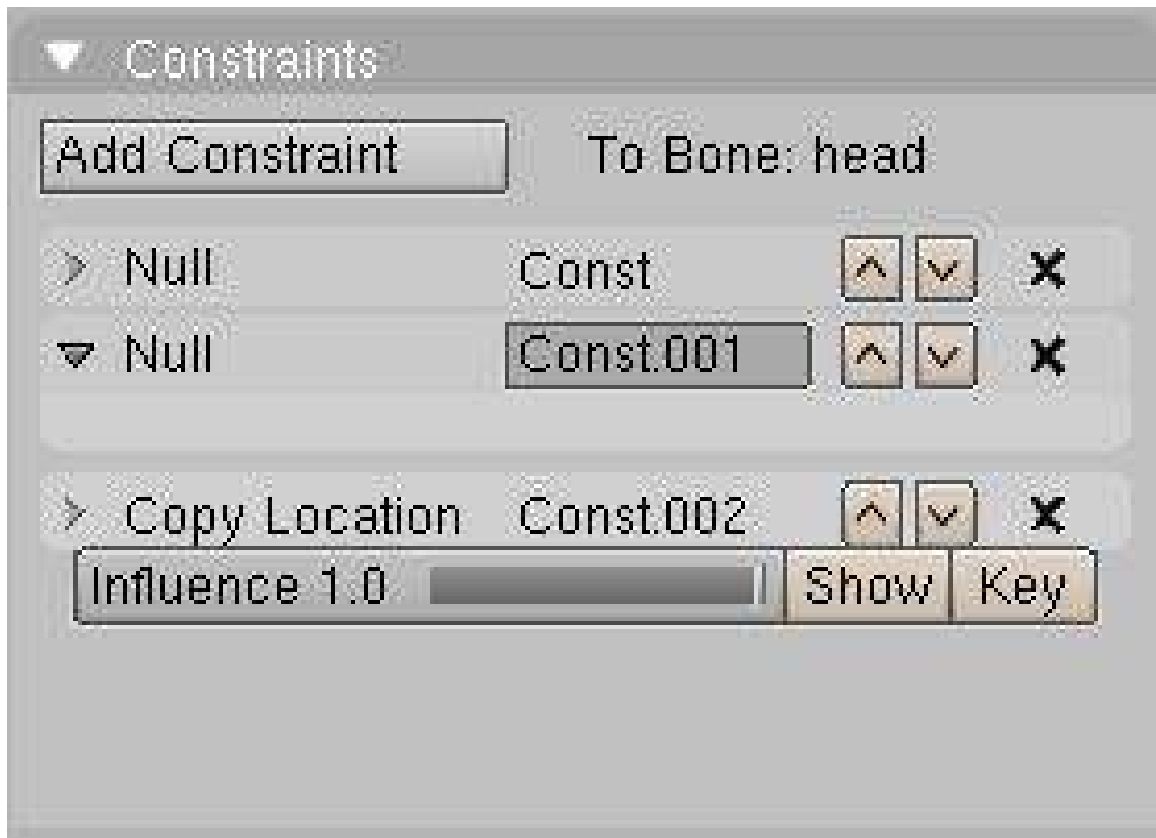


Figure 1048

The panel also appears in Editbutton(F9) when you are in Armature Editmode or Posemode. So what you get:

- A button to add a new Constraint. The choice you have is listed down this page.
- When you add a new Constraint, A block gets added in the stack. The UI is almost the same as the Modifier Stack. Each block represents an entry. You can delete it with "X", move it up or down in the stack, close or open it.
- Constraints are evaluated from first to last. So if you have two Constraints working on the same channel, let say Location, The last one will most probably win the chance to move the object. But...
- Most of the constraints have an influence slider to determine how much it will influence the stack. If the last constraint has an influence of 0.5 it will mix the result with the one before.
- You can animate the influence of the Constraint by moving the time, changing the Influence slider and adding a key with the **Key** button.
- The **Show** button will display the influence IPO curve in an IPO window for editing. (The IPO window must be opened before pressing the 'show' button).
- You can change the name of the Constraint by clicking on the name when the constraint is open.
- By Clicking on the white jacket of the Constraint you select which one is active for edition, same as "show" button.
- If most of the Constraint you can enter the name of the Object you want to work with as a target or reference. For a bone, you need to enter in which Armature object it is, then an other field for the bone name will appear. When filling those fields, remember you can use autocompletion using **TAB**.

162.3. The Constraint Index

- Copy Location¹
- Copy Rotation²
- Track-To³
- Floor⁴
- Locked Track⁵
- Follow Path⁶
- Stretch-To⁷
- IK Solver⁸
- Action⁹

1 Chapter 163 on page 1229

2 Chapter 164 on page 1233

3 Chapter 165 on page 1235

4 Chapter 166 on page 1239

5 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FAdvanced%20Tutorials%2FAdvanced%20Animation%2FGuided%20tour%2FConst%2Flt>

6 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FAdvanced%20Tutorials%2FAdvanced%20Animation%2FGuided%20tour%2FConst%2Ffp>

7 Chapter 167 on page 1241

8 Chapter 168 on page 1243

9 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FAdvanced%20Tutorials%2FAdvanced%20Animation%2FGuided%20tour%2FConst%2Fact>

163. Copy Location

163.1. Copy Location

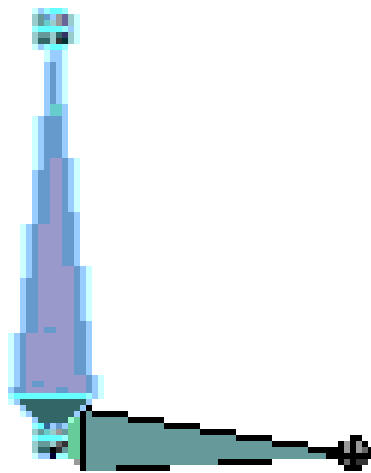


Figure 1049

The Copy Location constraint does as the name states: it will copy the location of the target to the source (constrained object). Less Influence will drag the constrained object less and less to the target.

If it's an armature, a new field will appear to let you tell which bone will be the target. Don't forget TABKEY completion while writing the name of your object/bone!

You can tell Blender to work only on the selected axis. Many uses are possible :)

163.2. The Constraint Panel



Figure 1050

- The Target field will let you select which Object the constraint holder will follow.

163.3. Where To Use It

Most of the time this little constraint is useful to stick objects to one another. By playing with the Influence you can tell when it will work, when it will remain motionless.

A good use of it is to ask a character to pick up something. By having a bone or empty for each side of the relationship (hand <-> glass), as the hand approaches the glass, you can align the two empties and fire the constraint up (1.00) to stick them together. You add another child-bone in the middle of the hand to tell where the glass will be. Thus moving the hand will move the glass. On the side of the glass just add an empty and make it parent of the glass. Add a copy location to the empty pointing to the bone in the hand we just did. There you go. Of course when the hand rotates the glass will not. For that you will need to add a Copy Rotation Constraint.

Before Blender 2.40, the above method was a good way of faking parent relationship without rotation. But now we have the hinge option which does the same.

Create this kind of tracking device using the X Y Z toggle button

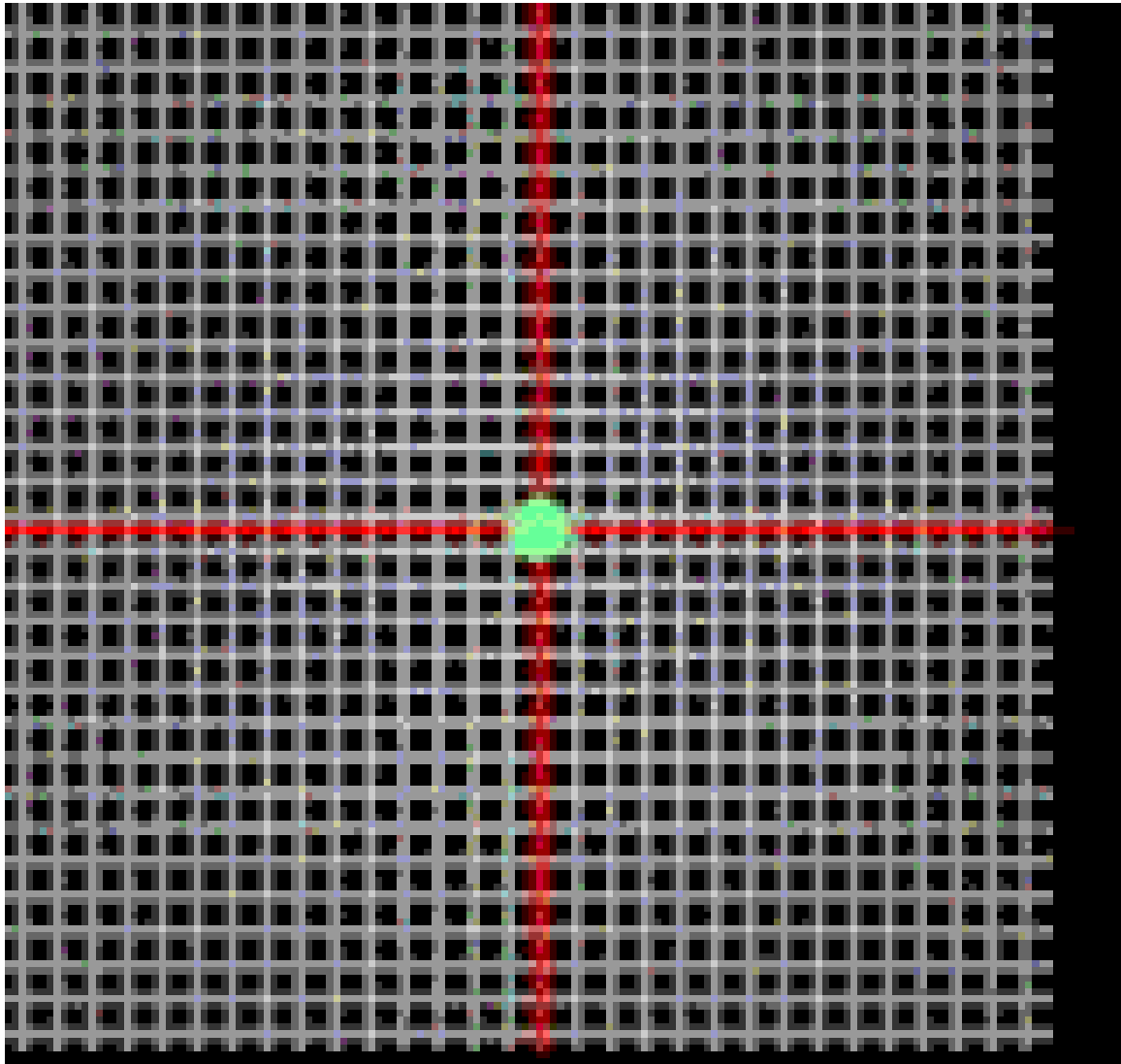


Figure 1051

164. Copy Rotation

164.1. Copy Rotation

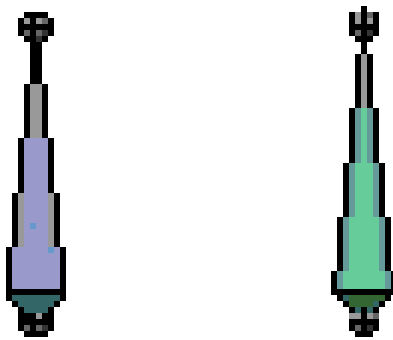


Figure 1052

This constraint copies the rotation of the target. As simple as that. It can be an object or a bone. As you can see in the example, only the rotation gets copied.

164.2. The Constraint Panel



Figure 1053

- You have 3 buttons to select which axis get copied over.

164.3. Where To Use It

Can be used with Copy Location to fake parent relationship. As you can key the influence you can make a character pick something up and hold it in his hands. Check the .blend for the hand-glass scene.

(?)link to the Blend(?)

You can also use this to align a plane with a 2D effect on it to the camera at all times. This works better than pointing it at the camera in some cases, such as a ring of atmospheric halo around a planet, where you don't want it disappear behind the planet.

165. Track-To

165.1. Track-To

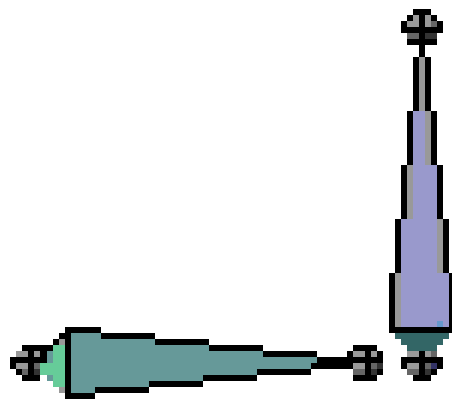


Figure 1054

The Track-To constraint lets you influence the Rotation of the constrained object by making it track a target with one of the constrained object's axis.

165.2. The Constraint Panel



Figure 1055

- You can enter the name of the target you want to track.
- You can select which axis is going to track the target.
- You can select which axis is going to stay up.

165.3. Where To Use It

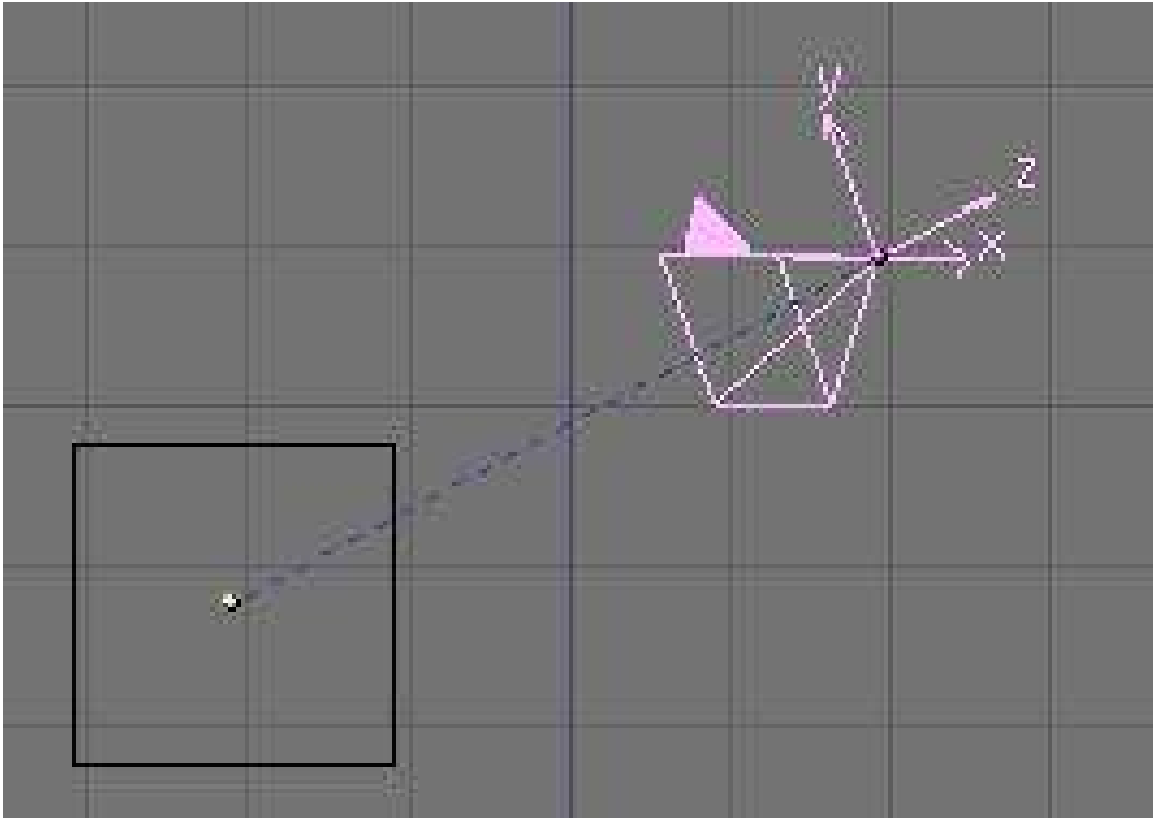


Figure 1056

A good example of use is to make a camera track an object. The setting to use on a camera is track: -Z and up: Y. You can turn Axis drawing in objectbutton window to help you choose the good axis.

Another example with armature would be the eyes of a character:

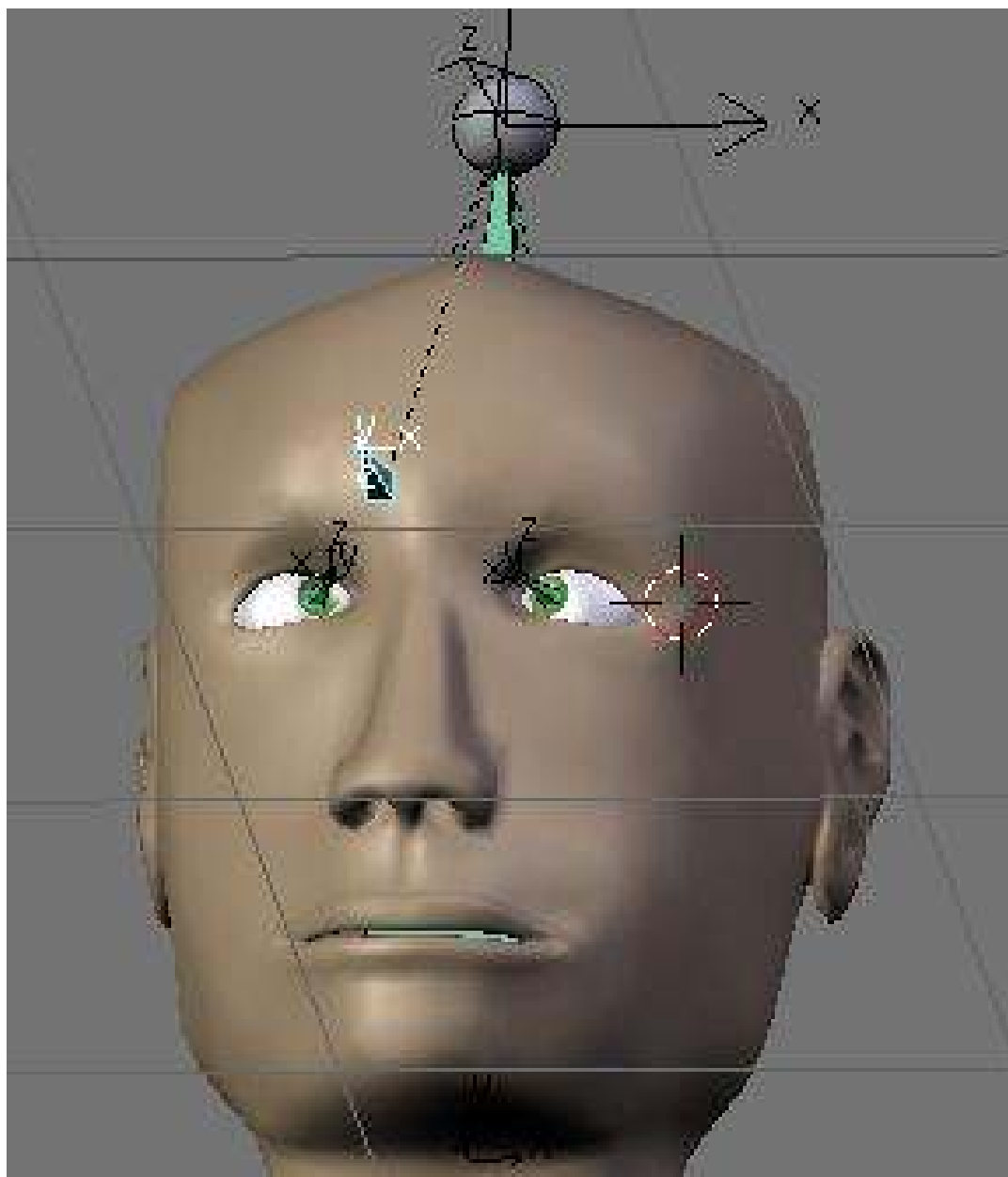


Figure 1057

(?)link to the Blend(?)

166. Floor

Using the floor constraint will keep a bone from passing through an object from a given direction. It is useful when making floors of course but also when making walls and other items which are obstacles for the armature.

There is also an offset value which is very useful when say for example you have a foot where the mesh stretches down below the actual tip of the armature you can then use the offset to make the bone stop before it actually reaches the obstacle object.

167. Stretch-To

167.1. Stretch-To

Description

Stretch To causes the affected object to scale the Y axis towards a target object. It also has volumetric features, so the affected object can squash down as the target moves closer, or thin out as the target moves farther away. Or you can choose not to make use of this volumetric squash-'n'-stretch feature, by pressing the NONE button. This constraint assumes that the Y axis will be the axis that does the stretching, and doesn't give you the option of using a different one because it would require too many buttons to do so.

This constraint affects object orientation in the same way that Track To does, except this constraint bases the orientation of its poles on the original orientation of the bone! See the page on Tracking for more info. Locked Track also works with this constraint. Options

R

Pressing the R button calculates the rest length as the distance from the centers of the constrained object and its target

Rest Length

Rest Length determines the size of the object in the rest position

Volume Variation

Volume Variation controls the magnitude of the effect

Vol

The Vol: buttons control along what axes the volume is preserved (if at all)

Plane

The Plane buttons define which local orientation should be maintained while tracking the target

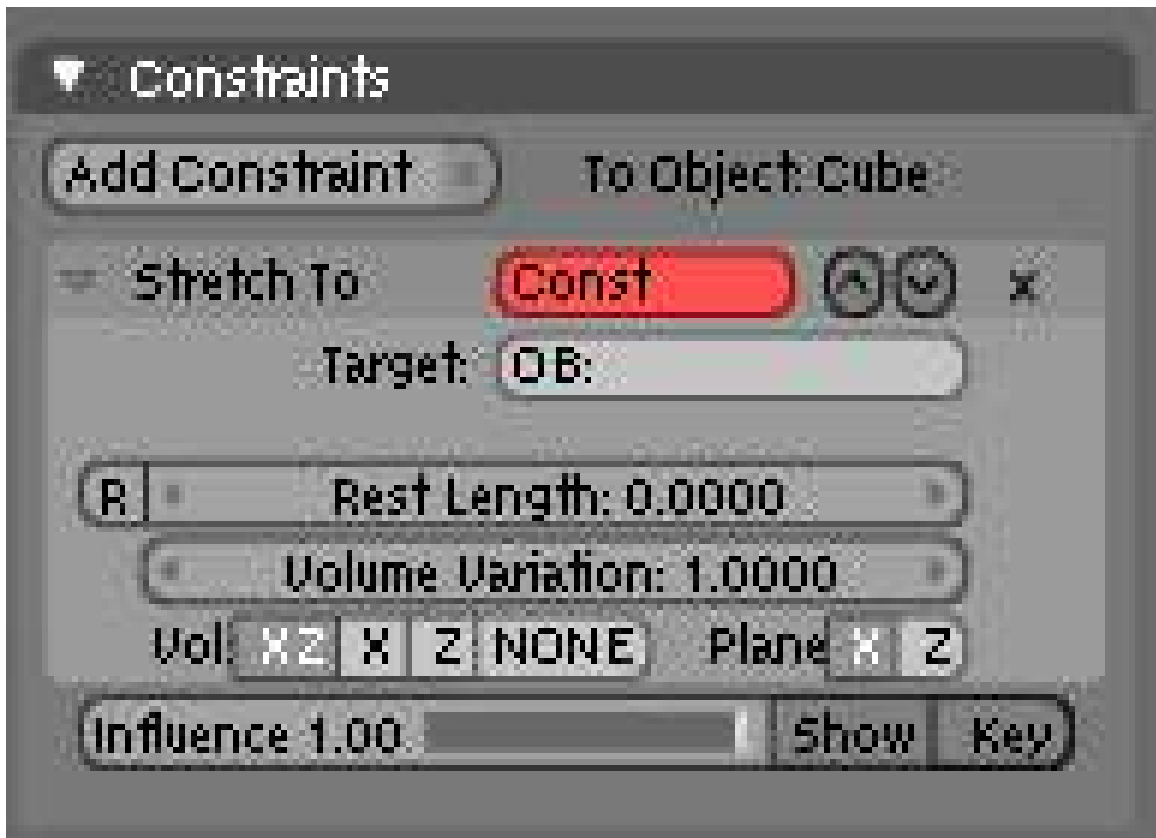


Figure 1058

168. IK Solver

168.1. The IK solver

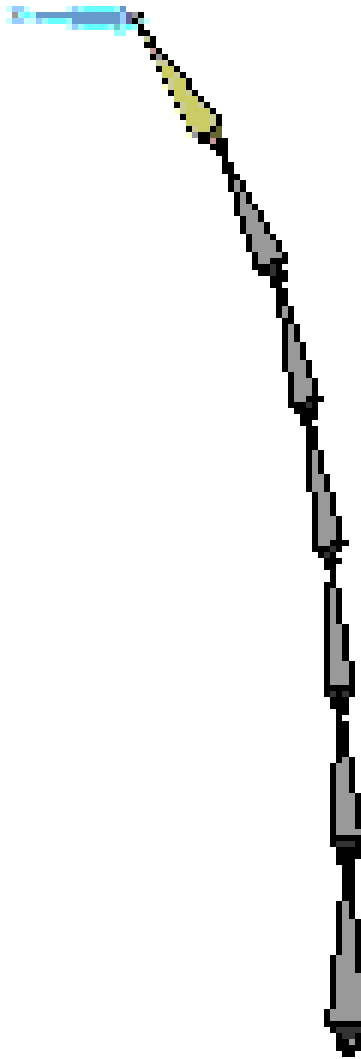


Figure 1059

The IK solver constraint is a wonderful tool for all animators. IK stands for "Inverse Kinematic" and is the opposite of FK (Forward Kinematic).

- FK: You have a dependency to the root of the chain. In Blender, a FK chain is a chain of bones connected together. You have to animate the bones rotation one by one to get it animated. It takes longer, but gives you entire control over the rig.
- IK: Both ends are roots. All bones in the chain are trying to rotate to keep both ends on targets. Now this Constraint got most of the attention during Animation refactoring, hopefully we have a lot of toys to play with now.

The IK solver has a special shortcut **in Posemode** to be added easily to a bone. If you select a bone and press '**CTRL-IKEY**', You get a little menu asking for more info on the new constraint, the target: to a new empty object or without target. It's now possible to work without target. Though you have less freedom (no rot feature, difficult parent relationship).

You can also select the target and then the IK constraint holder and press **CTRL-IKEY**. With this way of selecting ensure that your target is selected, but the bone you want to apply the constraint to is active (the last one selected). The menu will then let you add a constraint to the current bone with a target. If the target would itself be part of the IK chain, you get an error message - so make sure the target bone is not connected to the bone you want to add the constraint to.

It's also possible to remove all IK constraints from selected objects or bones with '**ALT-IKEY**'.

Q: '**CTRL-IKEY**' doesn't seem to do anything

A: Either the 3D Window is out of focus (R-click in empty space to solve) or you're not in Pose Mode ('**CTRL-TAB**', selected bone will be magenta in color)

A: In the 2.48 version of Blender, the shortcut is '**SHIFT-I**'

168.2. The Constraint Panel

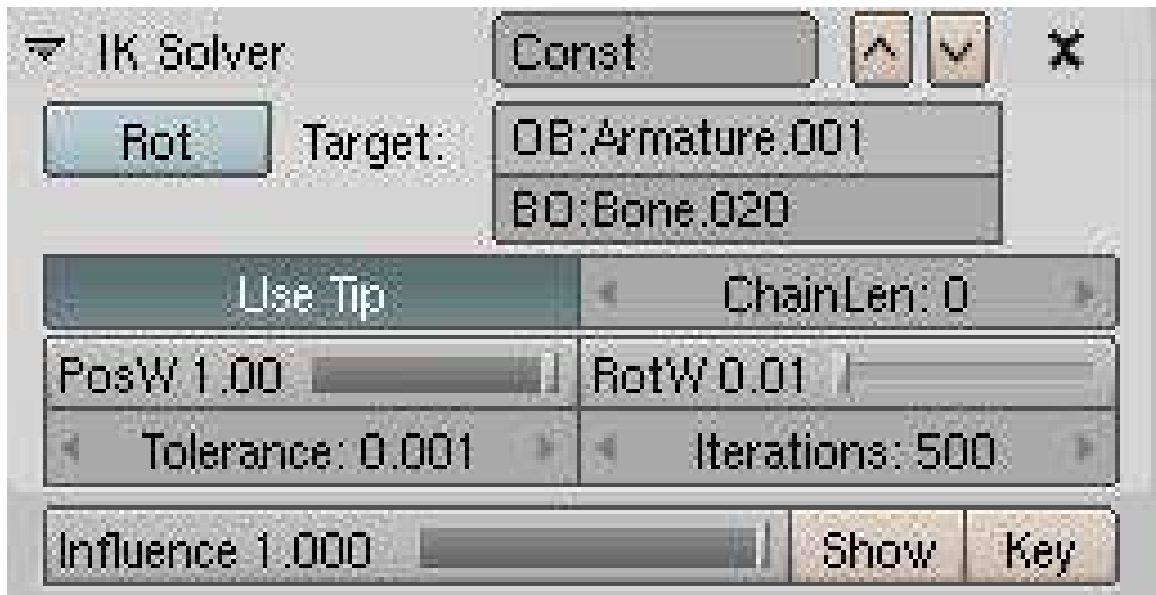


Figure 1060

- You can rename the constraint.
- You can select which Object or bone will be the target. Don't forget Tab completion.
- The Rot button let you tell Blender to use the rotation of the target to influence the rest of the chain:

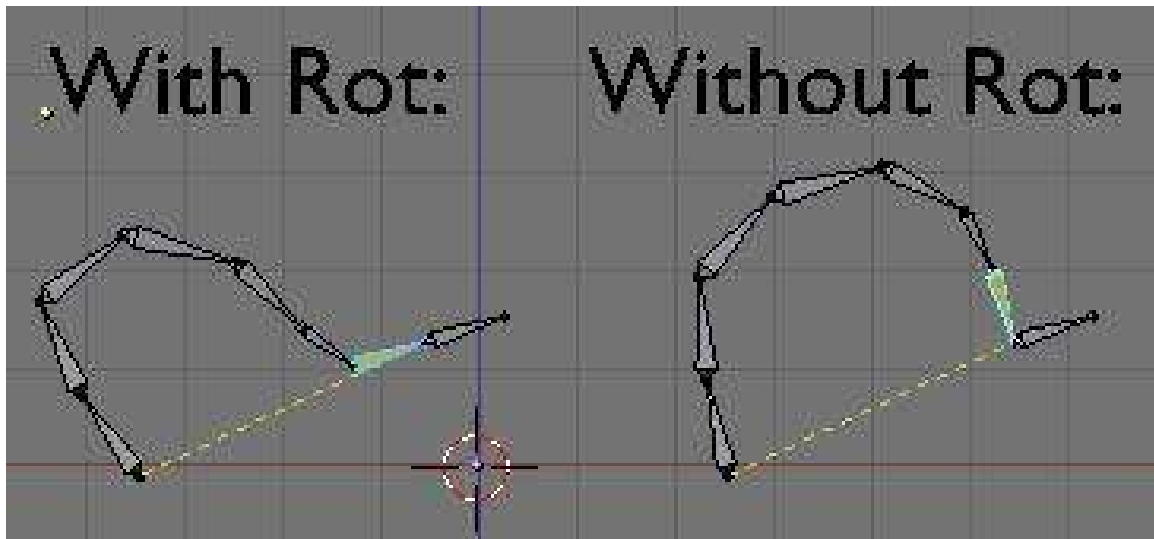


Figure 1061

- The Tip button lets you tell Blender which part of the bone is the target, the Tip or the Root. It's interesting to use tip, because this way the Bone holding the IK constraint can be used to deform geometry.

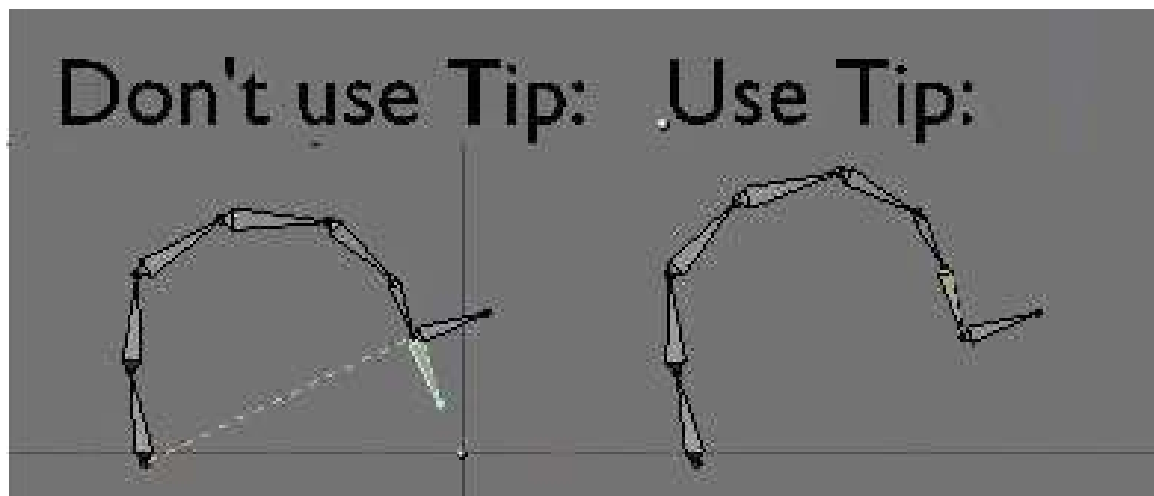


Figure 1062

- Len lets you tell Blender the length of the chain the IK solver will try to rotate. If set to 0, the entire chain will enter in the constraint. If for example the len is 4, only the 4 last bones of the chain will try to touch the target.

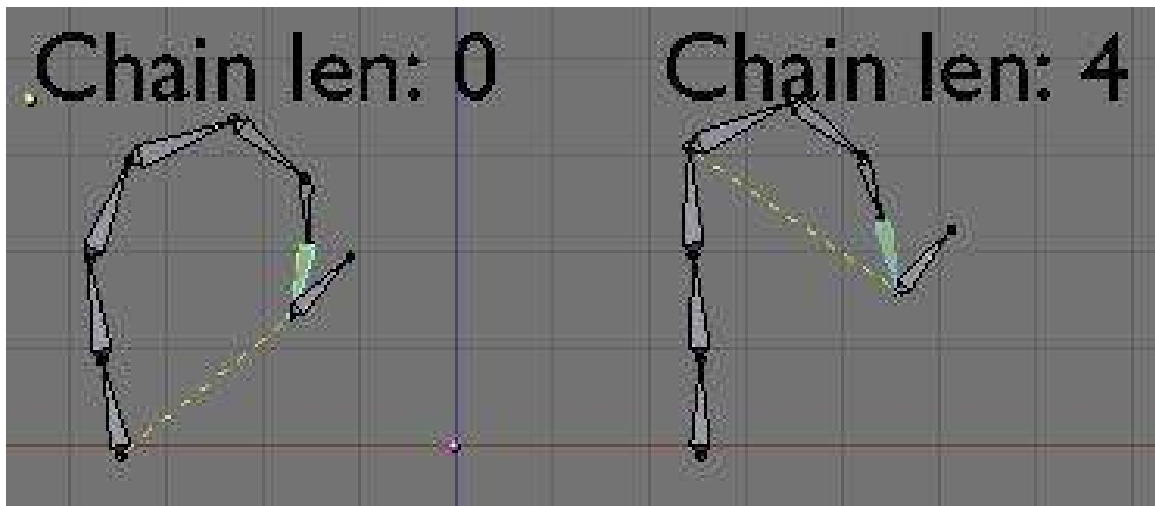


Figure 1063

- Also If you set len to 0 and your chain's root is a child of another bone, The IK solver will reach and rotate all the bones until it gets to the end of the parent relationship. If all the bones are linked up to a master root, then all other sub-branches will be affected. If there is another IK target in other sub-branches of the rig, Blender will try to mix them. This concept of multiple IK targets in a rig is called Tree IK and can be used to get completely automated animations. For example like a doll: if you pull one hand, all the body will follow. In the 3D-view you'll see a yellow line from the IK solver to the root of the chain it covers. This line appears when you select the bone containing the IK solver.

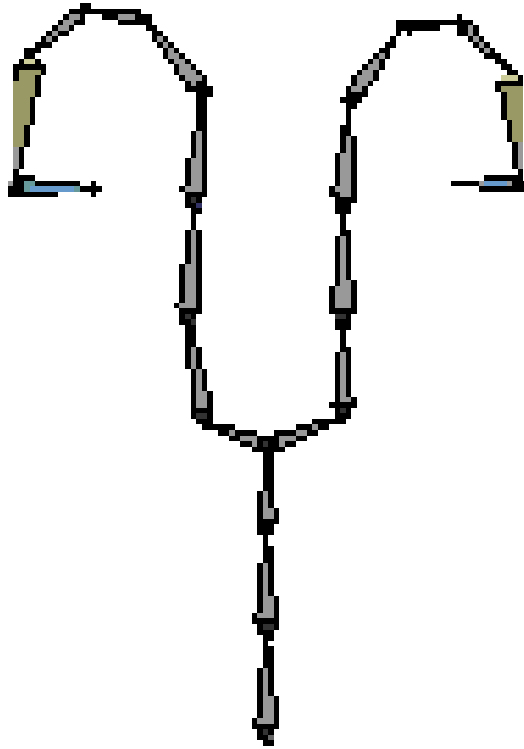


Figure 1064

- PosW and RotW let you tell Blender if this IK solver will influence the final result more or less in the case of a Tree IK setup. With these options it's possible to use an IK solver just for location and another one just for rotation.
- Tolerance and iterations are performance and precision options. IK solving is done in more than one pass, the more passes you calculate, the more accurate results you get. The tolerance is the distance from the IK solver to the target you can accept. If Blender manages to place the target near enough, it will stop doing iterations. The Iterations value is a hard limit you set to limit the time blender can reach on each IK solver per frame. Try to set it to a very low value to know why Blender needs more than one pass ;).
- You can set the general influence this constraint will have over bones, and it's animatable.

168.3. Where To Use It

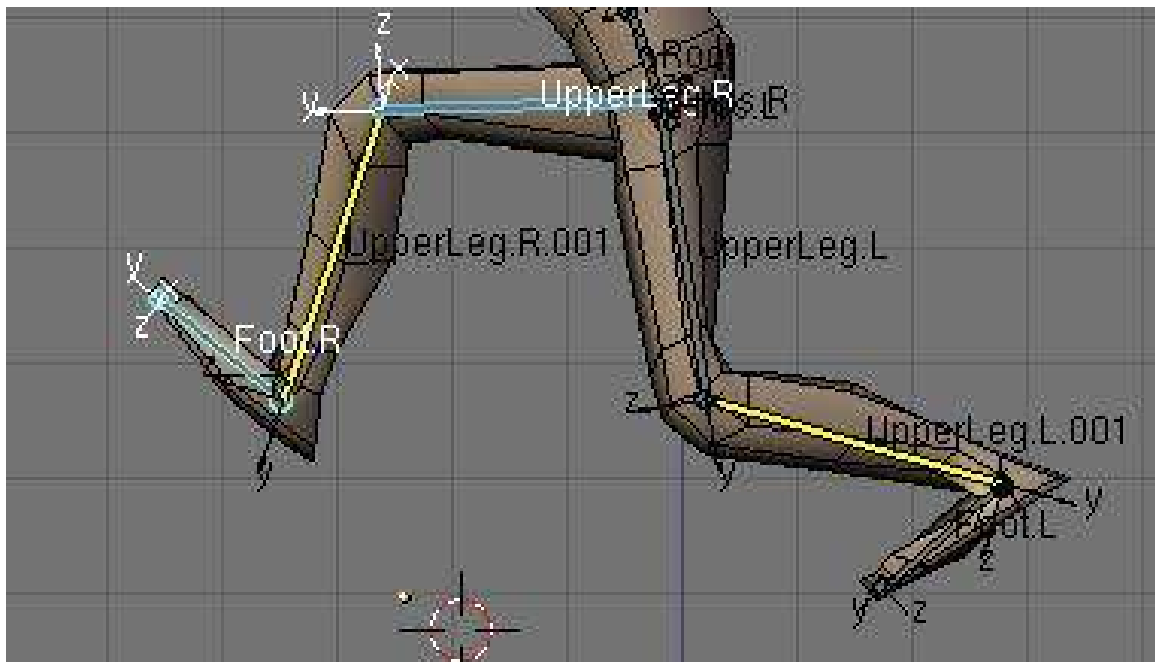


Figure 1065

In any chain of bones you don't want to animate by hand but you want both ends to be at precise location. The best example is a leg: The leg is connected to the body and to the foot. You don't need to animate the 2 bones in the legs, just place the body and the foot, the leg will follow automatically.

168.4. Degree Of Freedom

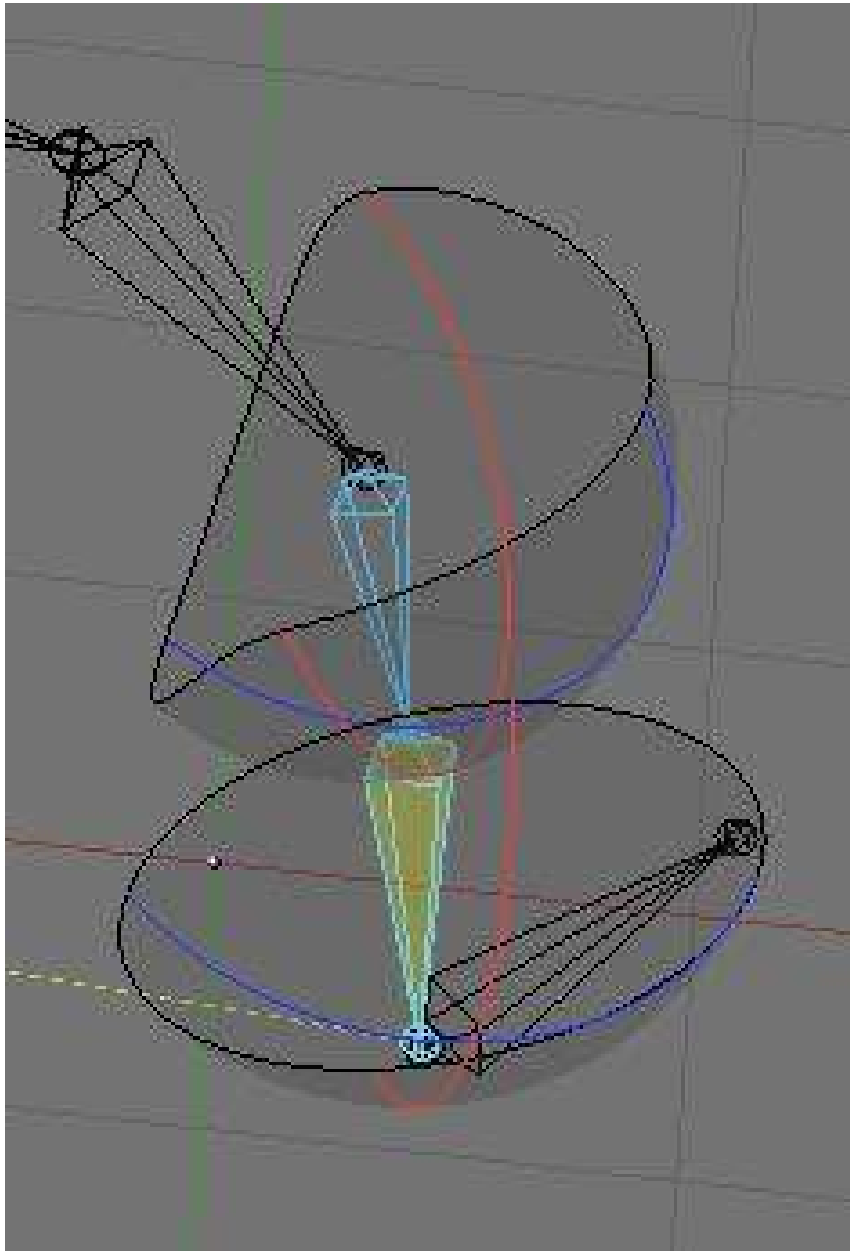


Figure 1066

In DOF it is now possible to set it for bones in an IK chain. This way you can set what will block where. This is very useful when doing a mechanical rig as you can limit the move or better, lock completely an axis.

Lock X Rot	Lock Y Rot	Lock Z Rot
Stiff X: 0.000		Stiff Z: 0.000
Limit X		Limit Z
Min X: -94.7		Min Z: -64.8
Max X: 118.4		Max Z: 88.6

Figure 1067

- There you can set a limit on each axis, or completely Lock it.
- No limit gives it complete freedom (which is the same as [min:0 max:360] or [min:0 max:0]).
- The stiffness lets you tell Blender if an axis is more difficult to rotate than the rest. If all bones have a stiffness of 1 on X and you try to curve that chain in a way that all bones need to turn on X to follow the target, the Solving will find really weird poses to find a way to touch the target without rotating on X.

169. NLA Window

170. (about this placeholder)

Since there has been nothing written on this page for a while, I asked for some basic information on another forum. I'll just quote what I learned from that forum, and this can be massaged into a real page of documentation over time. I won't attribute the individual various quotes, but thank you all for helping out. Remember, ANYONE can edit a Wiki, so create an account and make this thing better.

171. The NLA Window

171.1. Forum Notes

It's quite easy and maybe that's why there's no specific tutorial. Let's say you want to make two actions, AC:Hit and AC:Kick. Start with posing Hit and an Action will automatically be created in the Action Editor consisting of all the Bones that use Action IPO's. That's done so return to Frame 1 which will be your default Stance of AC:Hit.

Now, in the Action Editor, click the X (delete) next to AC:Hit and the datablock menu will disappear. (If you Add New instead of deleting then it will copy selected bones to the new action and you don't always want that). If you want a new default pose for AC:Kick, then Pose it or the same stance will be used as was the default in AC:Hit. Pose and Keyframe your Kick action and name it.

Over in the NLA Editor you can now use Shift-A to add NLA-Strips of your Actions, Grab and Scale them and use the Transform Properties tab to input how they Blend.

If you select an Action with the dropdown menu, at first its name will appear in the NLA window along with its keys. To make this Action into an NLA strip, point at the Action's name in the NLA and press CKEY.

Close any open Actions by clicking the scary X in the Action Editor. If you don't do this, only this action will play. Now in the NLA editor, play the animation.

If you see any keys (diamonds) in the NLA window, instead of strips (rectangles), you're still editing an action. It's so much easier if you have both the Action and NLA windows open so you can see whether an Action is open or not. (Edited by CD38 23 Feb 2006)

171.2. Walkthrough

With no Actions selected, both the Action Editor and the NLA Editor appear empty. Here, the NLA Editor window does list one Object called **-Empty** because that object is not an armature but it has some IPO curves attached.



Figure 1068 empty action window

Select an Action you've already made. Here, an Armature named **Yui** has bones involved in a one-frame action called **-AtEase**.

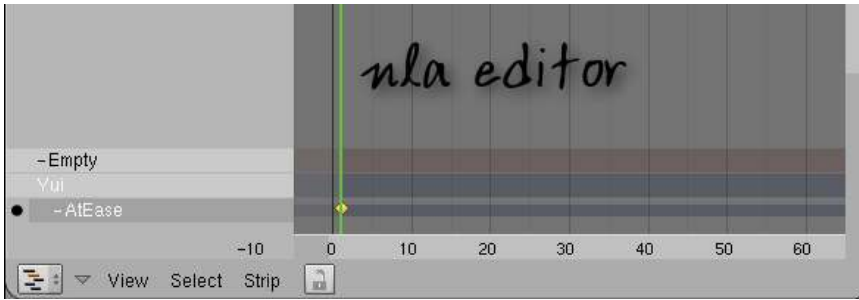


Figure 1069 nla window with action



Figure 1070 action window with action

Convert the listed Action to an NLA Strip in the NLA Editor by pressing the CKEY with the mouse hovering over the Action to be converted. No change in the Action Editor; it is still available as an Action.

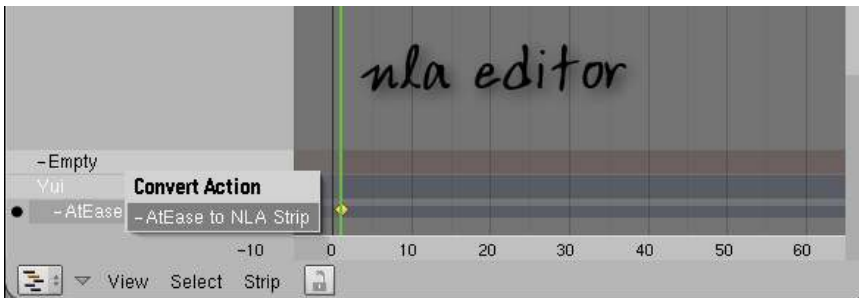


Figure 1071 nla window converting to a strip

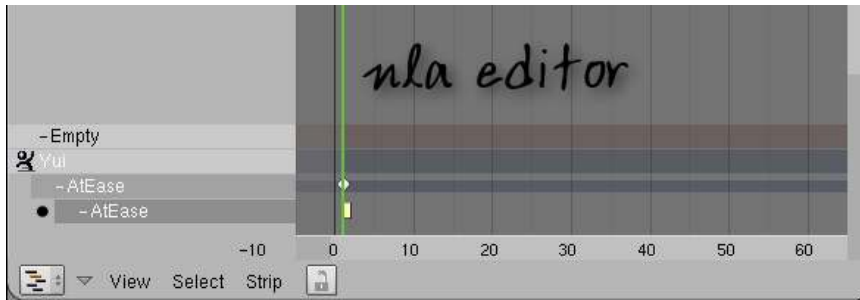


Figure 1072 nla window after conversion

Once converted, note the changes in the NLA Editor. The Action icon appears next to the Armature's name:



Figure 1073

. This is actually a button though it does not look like it, and you can toggle it between the Action and NLA Strips icon by clicking on it:



Figure 1074

.

More soon.

172. Intro to the NLA Editor

172.1. NLA (Non-Linear Animation)

WARNING-This page assumes the reader to understand the IPO window and the Action Editor window, as well as rigging a character with an armature object. This tutorial will make little sense without this previous knowledge.

Imagine this--yourself, sitting at your computer late at night, beating away at a huge blender animation. You didn't think it would take as long as it did, but you lost your composure staring into the jungle of colorful IPO lines, little white and yellow Action diamonds, and that annoying green current frame line. You know that the NLA window would help you make sense of it all, but you are afraid of opening that Pandora's Box because of the problems that will follow it. Never fear, for this tutorial will clean up all those problems and revolutionize the way you blend for the rest of your life.

172.2. Setting up the Scene

It will be easier for you if you start with a small demo file than if you go straight for the big prize. Give a character or other armature-rigged object a few SEPARATE actions (remember to name them something distinct, like "Walk" or "Run" as you always should with *everything*). For your own sanity, you will want to have a path or an IPO that correlates with the actions in question. All of the blender screenshots I will be using come from a file with a very basic stick person, rigged with an armature skeleton. He has two actions; a normal walk cycle, and then another, hunched over one, as if to pass beneath a low ceiling. If you use this idea in your own test file (and I thoroughly recommend it) do yourself a favor and give them both the same stride length! Give this guy an IPO or a path to follow that keeps his feet from slipping, and make sure that it has a linear interpolation mode. The rest of the scene consists of a floor-plane and an arch too small for him to walk under (hence the crouching walk). Place this arch exactly one walkcycle away for now, we will move it around later.

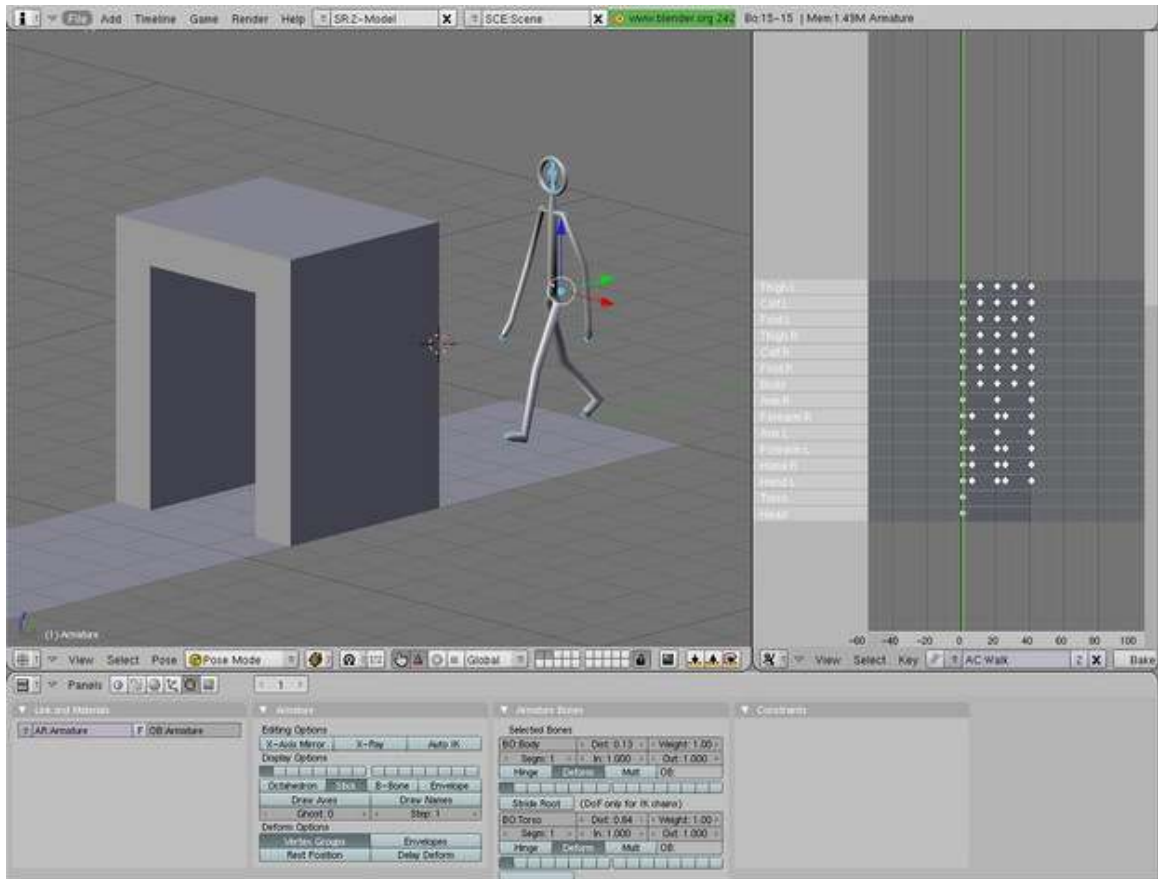


Figure 1075

Complete the scene before going on to the next step.

172.3. Adding Action Strips

Now, he moves forwards as you scroll through the frames, but how do we get him to walk forward, duck, and walk under the arch, too? First of all, you need to select the armature object and create a link to the normal walk. By pressing the up or right arrows, or pressing "alt+a", you should be able to see him walk up to the arch, stop walking, but keep sliding through, with his head sticking out the top. As ridiculous as this seems, right now, however, you are on the right track. Split your window now and open the NLA window with



Figure 1076

At first glance, this window looks almost identical to the action window we all know and love. This is not totally off of the mark. The NLA window is, in essence, an abbreviated version of the action and IPO windows.

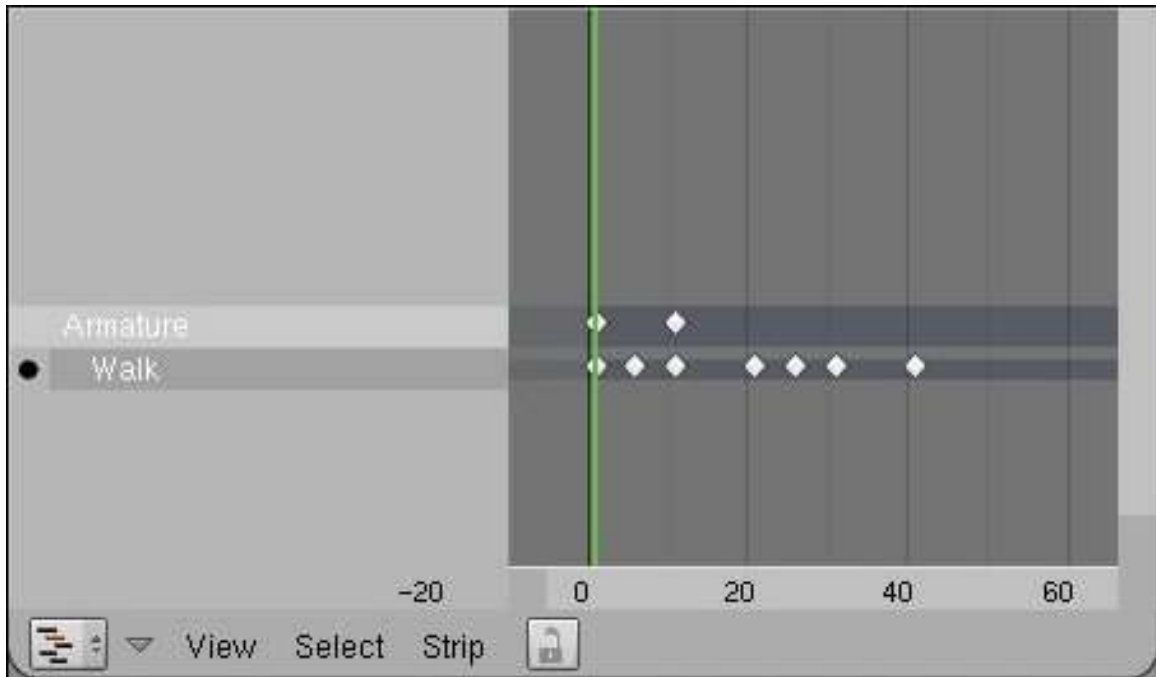


Figure 1077

This is what you should see in the NLA window.

You see a space with the name of your armature (named "Armature" in my demo) and in that row you see a few key diamonds. These show the ipo you have put on the person. I made it so the IPO extended on forever, using the first ten frames as a guide. In a subset beneath the armature object, you see the name of an action. The bullet next to it means that the armature is currently linked to that object. In that row, you see key diamonds, and those correlate with the keys you put into the action in question. If you switch actions in the action window, you see that the NLA window also changes. But how do you get it to keep the first and play the second later?

Go to the first frame you want him to walk in (most likely number one) and place your cursor in the NLA window. Press "shift+a" to see a dropdown menu. Pick the name of the first walk cycle. You should see a new subset space appear, with the name of the first walk in it. This space should be occupied by an action strip. Two colors of action strips are yellow and pink, yellow meaning selected and pink meaning deselected, just like IPO vertices. This strip takes up the amount of frames the walk did. The NLA window has now saved your first action. Now to toggle to the next action, move the frame line to the end of the first strip, then press "shift+a" again, this time clicking the other action on the dropdown menu.

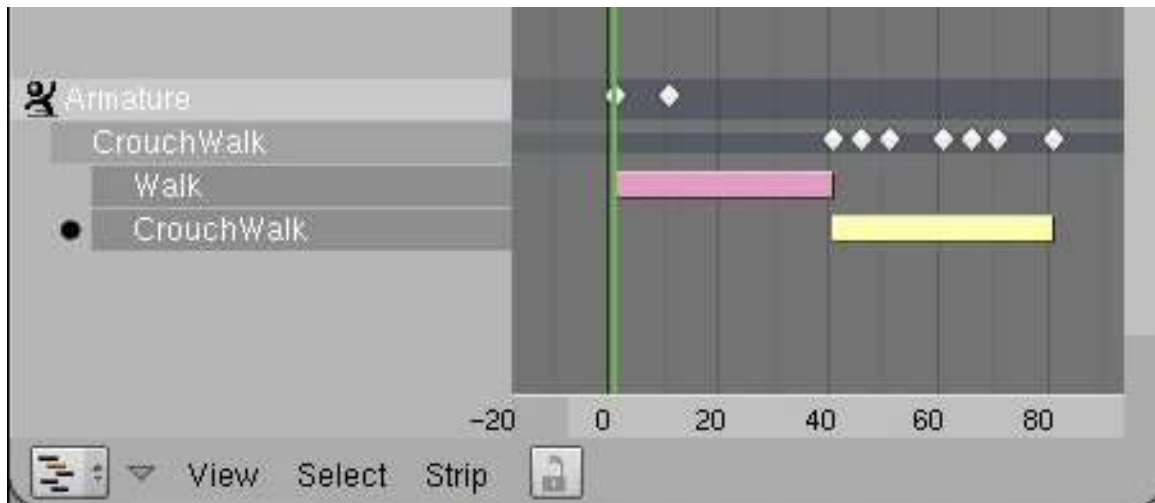


Figure 1078

The result.

172.4. Getting the Strips to Play

Now, as cool as this seems so far, if you press "alt+a", you will be disappointed and confused. The animation will only play one of the actions. This is a simple problem to fix. If you look carefully, you will see that the armature is still linked to the action it played. This data is overriding any other data from the NLA window. Press the "X" in the action window right next to the name of the action. Press "alt+a" again. It works!

This is, however, a jerky and instantaneous switch from one action to the other. Making it less weird is easy. Press the "N" key with your cursor in the NLA window. You will see a small box with a bunch of functions in it (this will be explained in detail on later pages).



Figure 1079

The Properties window.

This little box is the lifeblood of the NLA in blender. It contains all the tools you will need to be able to run this feature smoothly. I will illustrate a few of them right now, but most others will be shown in the section on this window.

Make sure you have highlighted the *second* action strip, and look at the space where it says "Blendin". The number there is the number of frames blender will use to smooth the transitions *in to* the highlighted action. Depending on how fast your character is walking, you may need to change the number a bit, but I set my "blendin" for seven. Whatever number you used, however, move the second strip *back* that same amount of frames. The wedge on the end of the action strip should end as the other strip ends.



Figure 1080

The NLA solution.

172.5. Conclusion

You now have the necessary skills to complete much more involved and complex animations. You can, however, increase this even more by continuing to read the other NLA tutorials in this wikibook.

Good Luck!

173. The Stride Feature

At the moment, one of the best documentation for Blender Stride features can be found here: [Blender Stride Tutorial](#)¹, which takes on where the [Official Blender Stride Page](#)² takes off. Good luck! A simpler and shorter, but still very good tutorial can be found here: [Blender wiki: NLA Editor and Stride Path](#)³.

1 <http://www.telusplanet.net/public/kugyelka/blender/tutorials/stride/stride.html>

2 http://www.blender3d.org/cms/Advanced_Stride_support.720.0.html

3 http://wiki.blender.org/index.php/NLA_Editor_and_Stride_Path#Stride_Support

174. Relative Vertex Keys

1. REDIRECT Blender 3D: Noob to Pro/Advanced Tutorials/Relative Vertex Keys¹

¹ <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro%2FAdvanced%20Tutorials%2FRelative%20Vertex%20Keys>

175. Piston, Rod and Flywheel

This page relates to Blender v.2.42a

175.1. Piston, Rod and Flywheel

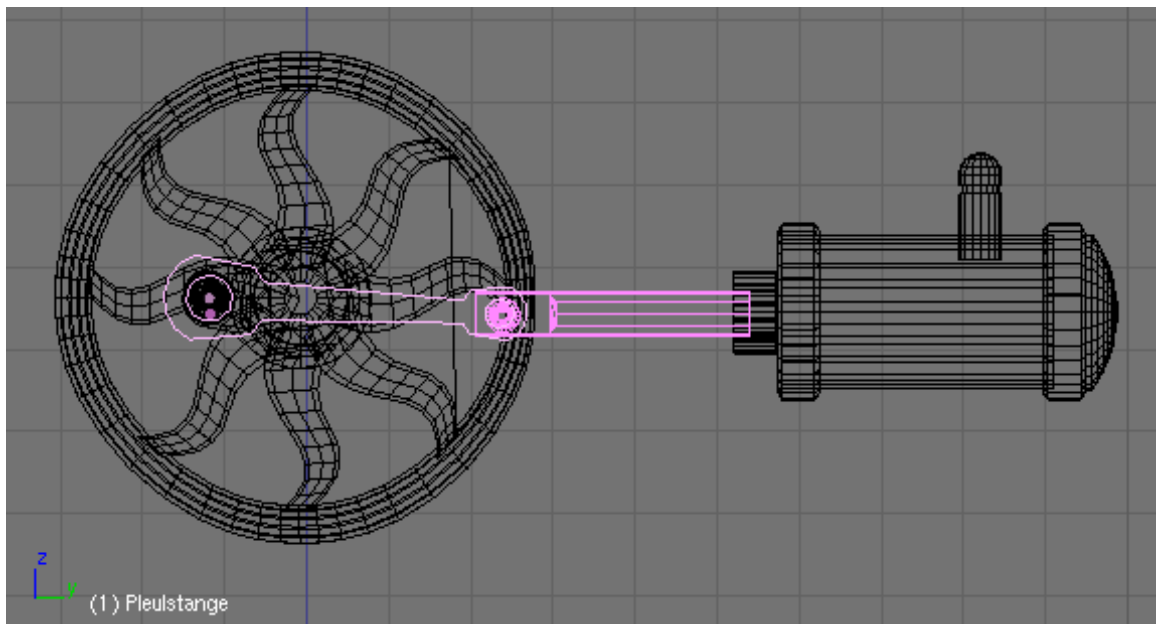


Figure 1081 Image 1: Surprisingly difficult: the animation of Piston [Kolben], Rod [Pleuelstange] and Flywheel [Schwungrad]

] A setup like that shown in **Image 1** is e.g. typical for a steam engine. It is surprisingly difficult to animate this setup, because one has to translate a rotating movement (of the Flywheel) to a horizontal movement (of the Piston). This alone wouldn't be too difficult, but the Rod has to stay attached to the Piston. The shown setup uses three empties and two armatures, each with two bones.

The horizontal direction is the Y-direction, if this is different in your model you have to exchange Y with your horizontal direction.

175.1.1. Setup

- Use **Ctrl-A->Apply scale and rotation** on all mechanical parts in object mode.

This will spare you from bad surprises if you rotate the flywheel afterwards.

We need an object which follows the rotation of the flywheel. We use an Empty, which is parented to a Vertex.

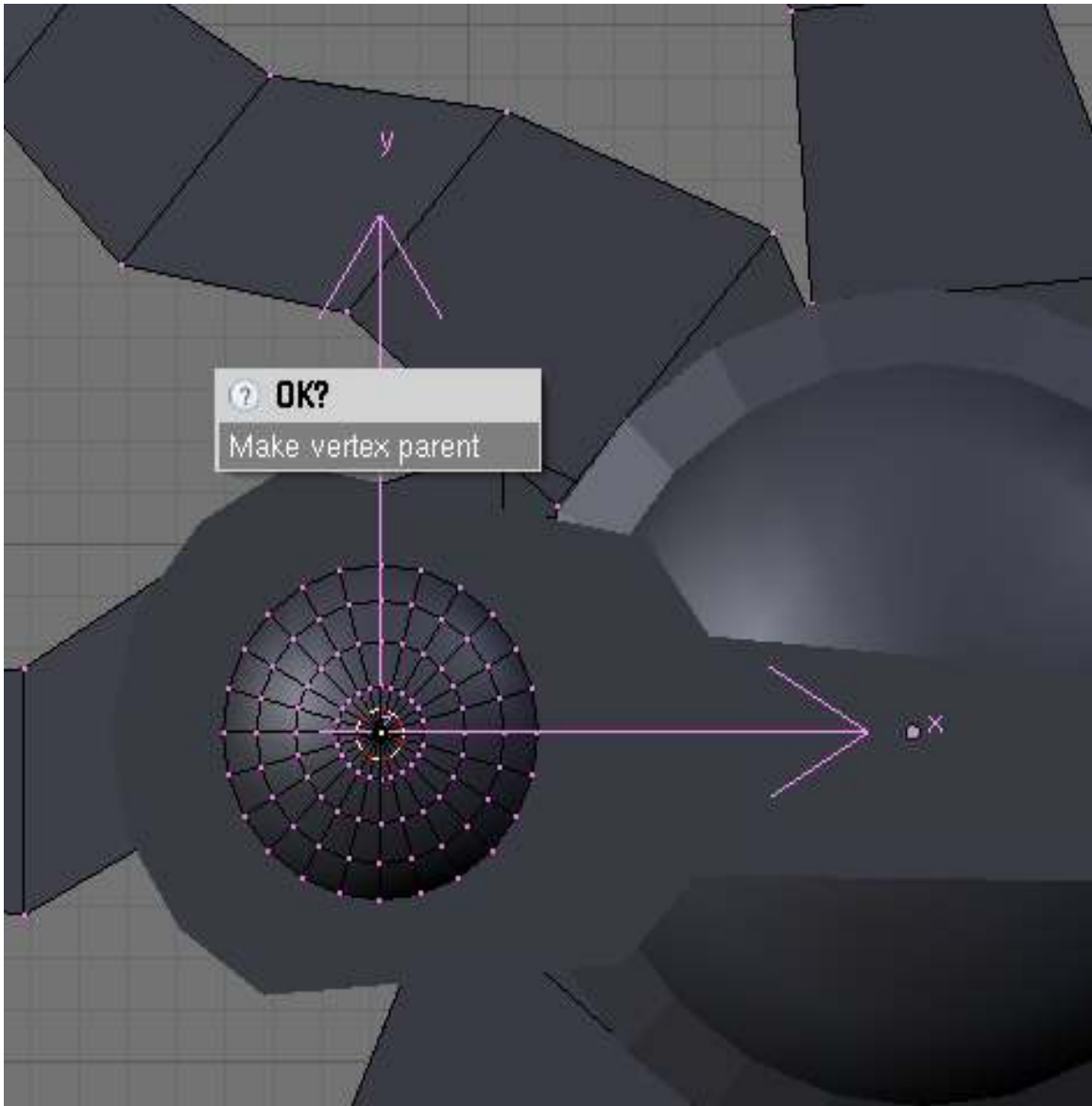


Figure 1082 Image 2: *Make vertex parent* of an Empty.

- Change to edit mode of the flywheel. Select a Vertex in the middle of the crank.
- Move the cursor to the Vertex (**Shift-S**->*Cursor->Selection*).
- Change to object mode and insert an Empty (*ERotation*).
- Change back to edit mode of the flywheel. The vertex is probably already selected, now select with **Ctrl-RMB** the Empty and press **Ctrl-P**->*Make vertex parent* (**Image 2**).

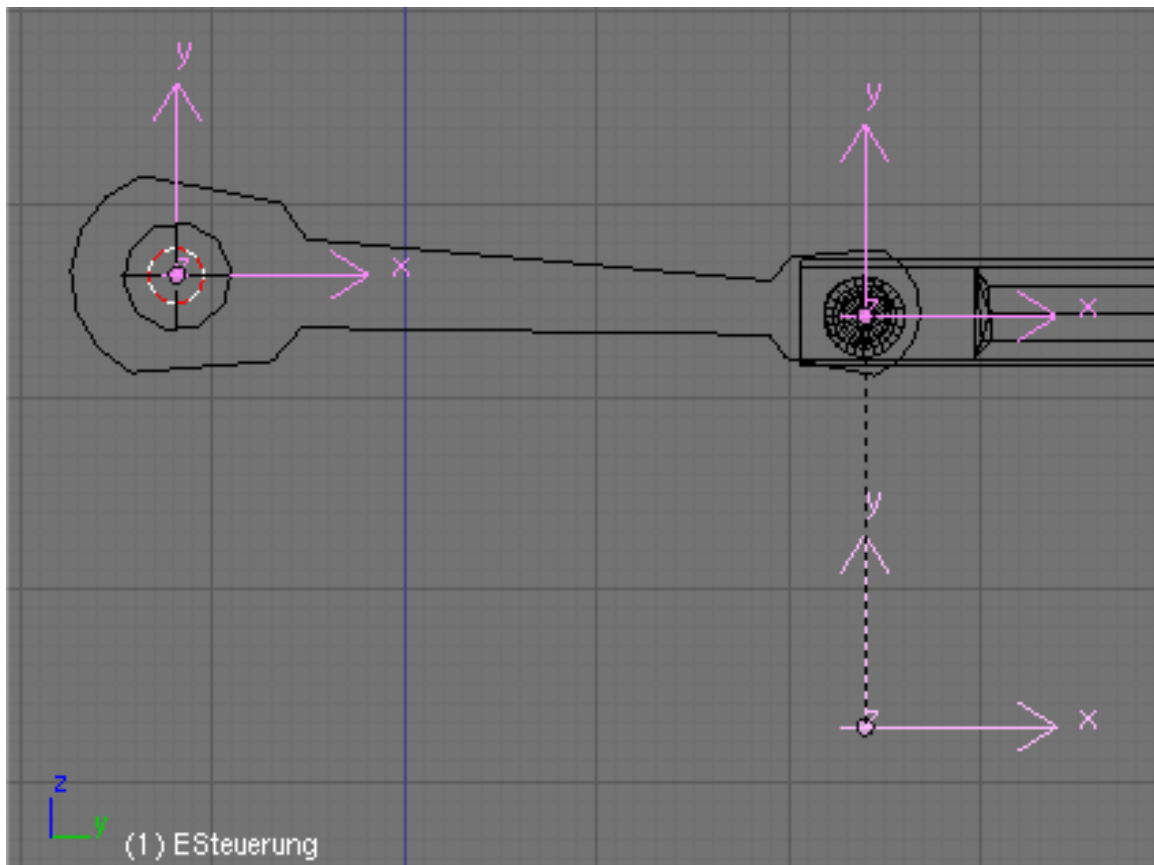


Figure 1083 Image 3: The positions of the three Emptys.

We need two more Emptys:

- Place the second Empty (*EGelenk*) in the center of the joint between Rod and Piston.
- Place the third Empty (*ESteuerung*) near to the joint.
- Now select at first *EGelenk*, than **Shift-RMB** *ESteuerung* and parent *EGelenk* to *ESteuerung* with **Ctrl-P**.

We're going to insert two armatures, each with two bones. The first armature, helps translate the rotation to a horizontal movement. The second armature drives the Piston and the Rod. The cursor should be positioned at the Empty *ERotation* (like in **Image 3**), else set it there.

- Add an armature (*AHelfer*). The root of the first bone (*BHelfer1*) has the same position as *ERotation*, the tip has to be positioned to the Empty *EGelenk* (select Empty first, *Cursor->Selection*, than select bonetip, *Selection->Cursor*).
- Extrude another small bone (*BHelfer2*) in horizontal direction.

BHelfer2 is automatically *Child* of *BHelfer1*. The second armature that is driving the movement of piston and rod will be inserted in the opposite direction, i.e. from right to left in our example.

- Position the cursor on the piston.
- Add an armature (*AGestaenge*).
- Move the tip of the first bone (*BKolben*) to *EGelenk*.
- Extrude the first bone and place the tip of the second bone (*BPleuelstange*) to *ERotation*.

Connecting the two bones will ensure that the rod cannot disconnect from the piston.

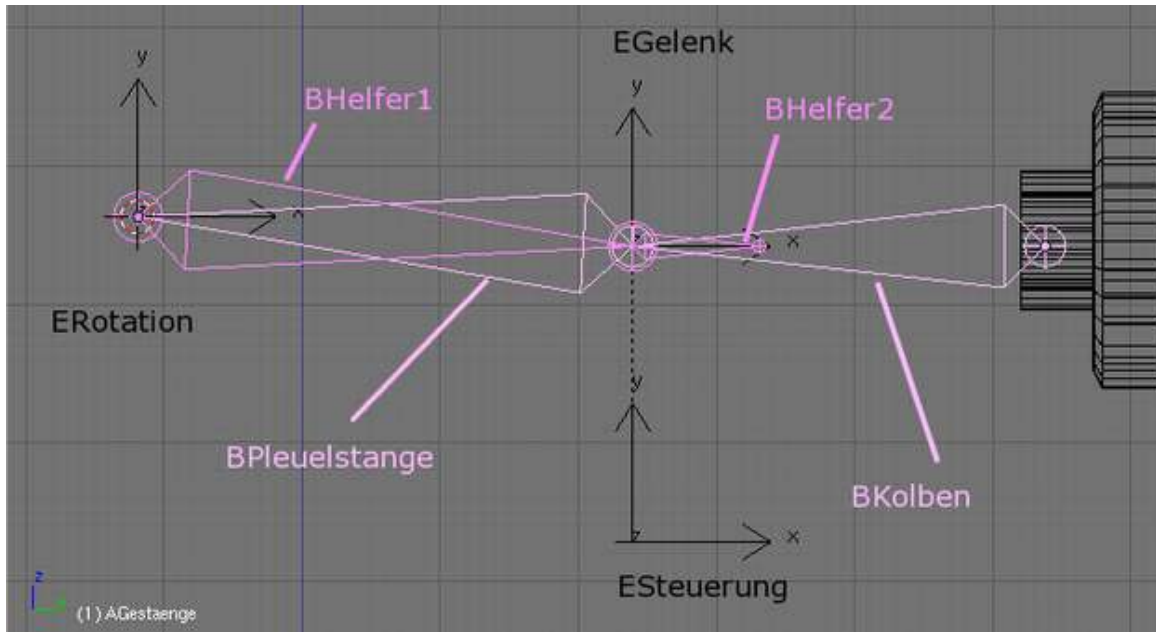


Figure 1084 Image 4: The alignment of the two armatures.

Now we're going to add the "logic".

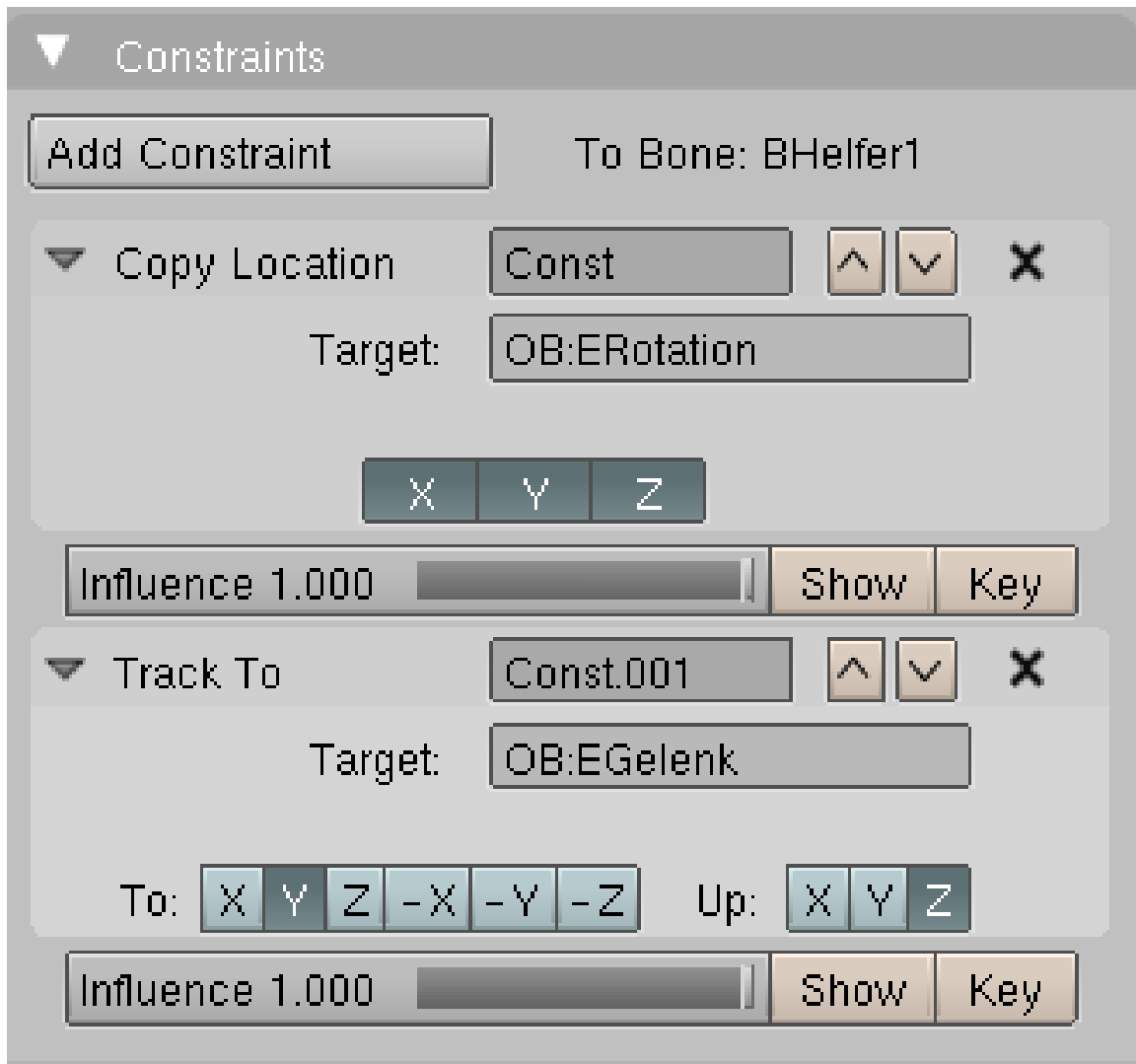


Figure 1085 Image 5: The constraint settings for *BHelfer1*.

- Add a *Copy Location* constraint to *BHelfer1* which points to *ERotation*. To do this change to the pose mode of the armature *AHelfer* and select *BHelfer1*. In the *Constraints* panel click *Add Constraint*->*Copy Location* and type *ERotation* into the *Ob:* field.
- *BHelfer1* needs a second constraint: *Track To* to *EGelenk* (**Image 5**).

Now the root of the bone *BHelfer1* follows the rotation of the Flywheel. It's tip follows the joint, but with slight deviations upwards and downwards. The position of *BHelfer2* governs the horizontal position of the Piston. We just have to "extract" the horizontal position, for that we're going to use a *CopyLocation* constraint, but only in Y-direction. There is no offset for a copy location constraint - else we could have done with one empty less.

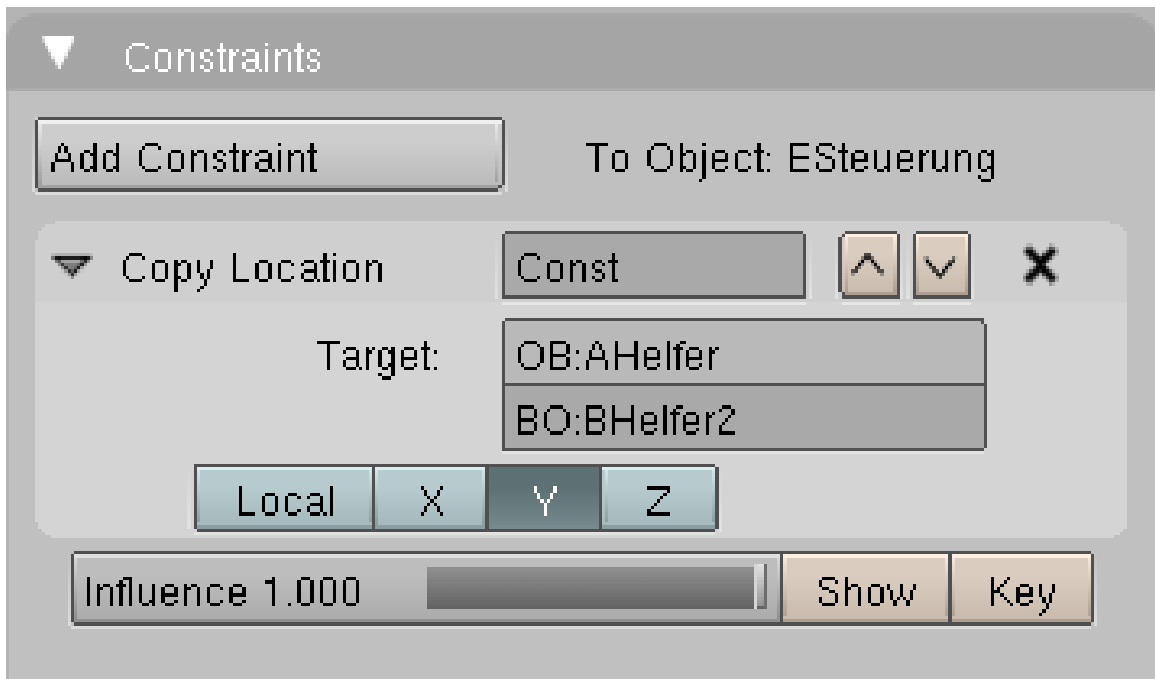


Figure 1086 Image 6: Constraints of *ESteuerung*.

- *ESteuerung* gets a *CopyLocation* constraint in Y-direction to *AHelfer*->*BHelfer2*. You have to inactivate X and Z (**Image 6**).
- Make *ESteuerung* the parent of *AGestaenge*.

The armature *AGestaenge* gets its horizontal movement from *BHelfer2* via *ESteuerung*.

Nearly finished:

- Give *BPlueelstange* a *Track To* constraint to *ERotation*.
- Parent the piston and the rod (the meshobjects) to their bones.

175.1.2. Overview about the relations between armature bones and objects

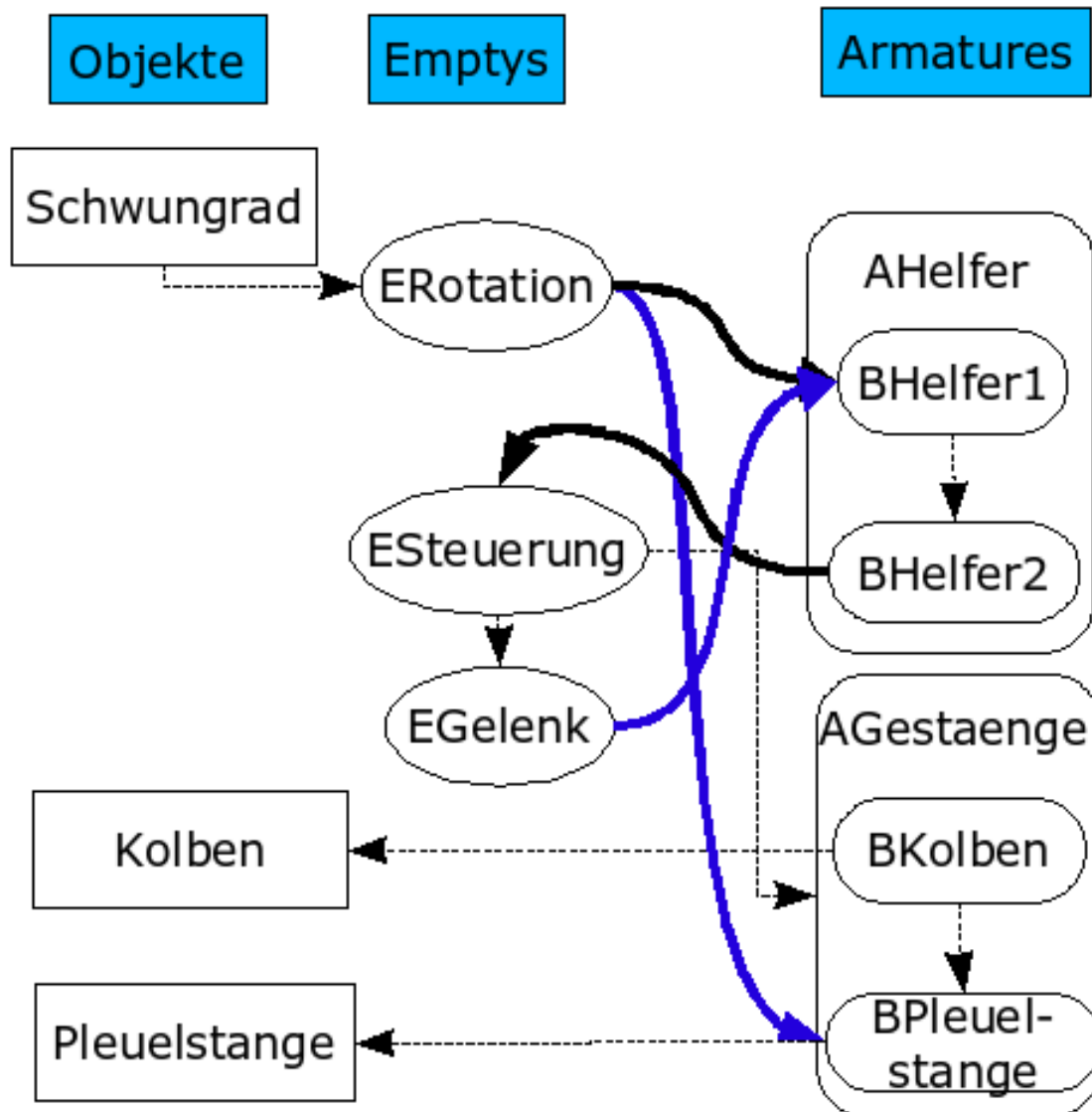


Figure 1087 Image 7: Parent-Child Relations are drawn with slim arrows, *Copy Location* Constraints are drawn with thick black arrows, *Track To* Constraints are drawn with thick blue arrows.

- *Schwungrad* is vertex parent of *ERotation*.
- *BHelfer1* has a *Copy Location X/Y/Z* constraint to *ERotation*.
- *BHelfer1* has a *Track To* Constraint to *EGelenk*.
- *ESteuerung* has a *Copy Location Y* constraint to *BHelfer2*.
- *ESteuerung* is parent of *AGestaenge*.
- *ESteuerung* is parent of *EGelenk*.
- *BPleuelstange* has a *Track To* constraint to *ERotation*.
- *BKolben* is parent of *Kolben*.

- *BPleuelstange* is parent of *Pleuelstange*.

The Parent-Child relations between the bones are inserted automatically through extruding, you have to insert the other relations yourself.

175.1.3. Finishing Touches

- Animate the rotation of the flywheel.

The driving object is the flywheel, all the other movements are synchronized with its rotation.

If you've done everything right, your animation should look like in **Image 8**.

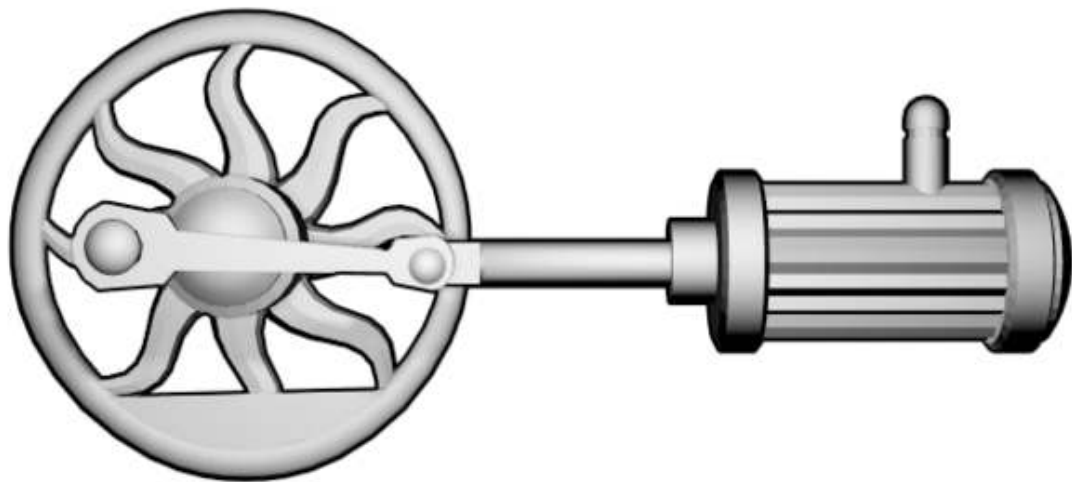


Figure 1088 **Image 8:** Our working Piston, Rod and Flywheel animation.

175.2. Links

The German original of this tutorial (same author)¹

¹ <http://en.wikibooks.org/wiki/%3Ade%3ABlender%20Dokumentation%2F%20Tutorials%2F%20Armatures%2F%20Kolben%20und%20Zylinder>

Another method using constraint system²

² <http://blenderartists.org/forum/showthread.php?t=140734>

176. Cutting Through Steel

This tutorial has been created with Blender v2.48a

Figure 1089 **Figure 1:** Cutting out a logo, the result of this tutorial

In order to simulate a laser beam which is cutting through a steel plate, we have in Blender (at least) two possibilities:

1. an animated Mesh with many Shapekeys
2. an animated texture

The latter method is considerably easier to implement and the result is nevertheless convincing enough. We will proceed as follows:

- You need an animated alpha mask for the points at which the metal is changed. For this use a text object as a path. A cube follows the path and emits a particle system. The results are rendered as animation and the images stored.
- Then load these pictures and use them as an animated texture on a plane.
- With a second particle system the sparks are simulated.
- With a third particle system we simulate the annealing of the weld. Again, this sequence is used as an animated texture.
- Smoke is created with a fourth particle system.

The result is shown in **Fig. 1**.

This guide is based on the well-known tutorial *Cutting Through Steel*¹, but remade with the current Blender particle system and with constraints. You need to have basic knowledge of Blender.

¹ <http://www.blendernation.com/tutorials/blender-3d-intermediate-tutorial-cutting-through-steel>

176.1. Creating the Alpha-Map

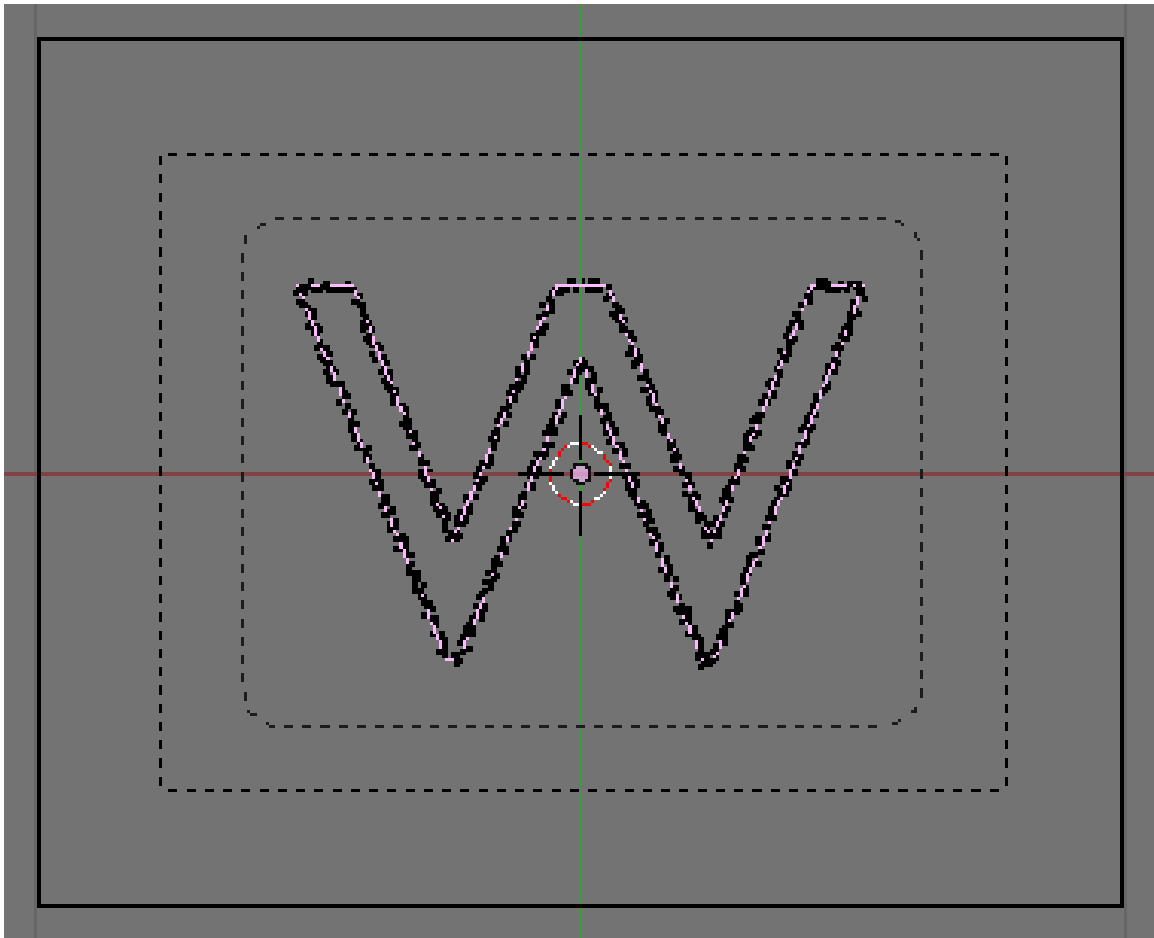


Figure 1090 Figure 2a: Setup of curve, emitter and camera in the 3D window.

- Start Blender. Save your file. Delete the cube.

We should save the file right at the beginning, because we will save the animated textures in subdirectories to the current directory. Particle caches are also created as subdirectories.

- Set the *End* value to 150 in the *Anim* panel of the *Render* buttons. This allows you to specify the total length of the animation.
- In the *Output* panel use `"//AlphaMap/"` as output directory. When we render the animation the images are then saved in the subdirectory "AlphaMap" to the current directory.

The animated alpha map consists of gray-scale images whose luminance information is used to calculate the mask. Bright pixels produce transparency, dark pixels remain opaque.

- Use *PNG* as output format. This avoids compression artifacts which occur with the *JPEG* format.

I've left the resolution at 800x600, but you need to set your desired resolution here.

- Create the shape you want to cut out as Curve object, or use a text object and convert this to a curve. I've done the latter, I've used a text-object (a "W"), and converted this to a Curve object (*Object-> Convert Object Type ...-> Curve*).
- With the function *Center* in the *Curve and Surface* panel I've brought the object in the middle of the 3D-coordinate space.
- The filling of the object is not needed here, therefore activate the option *3D* for the curve object, even though the object should remain two-dimensional.

Now we need again a cube. The cube will follow the curve. It's particle system will create a glowing trail, that is later used as alpha map.

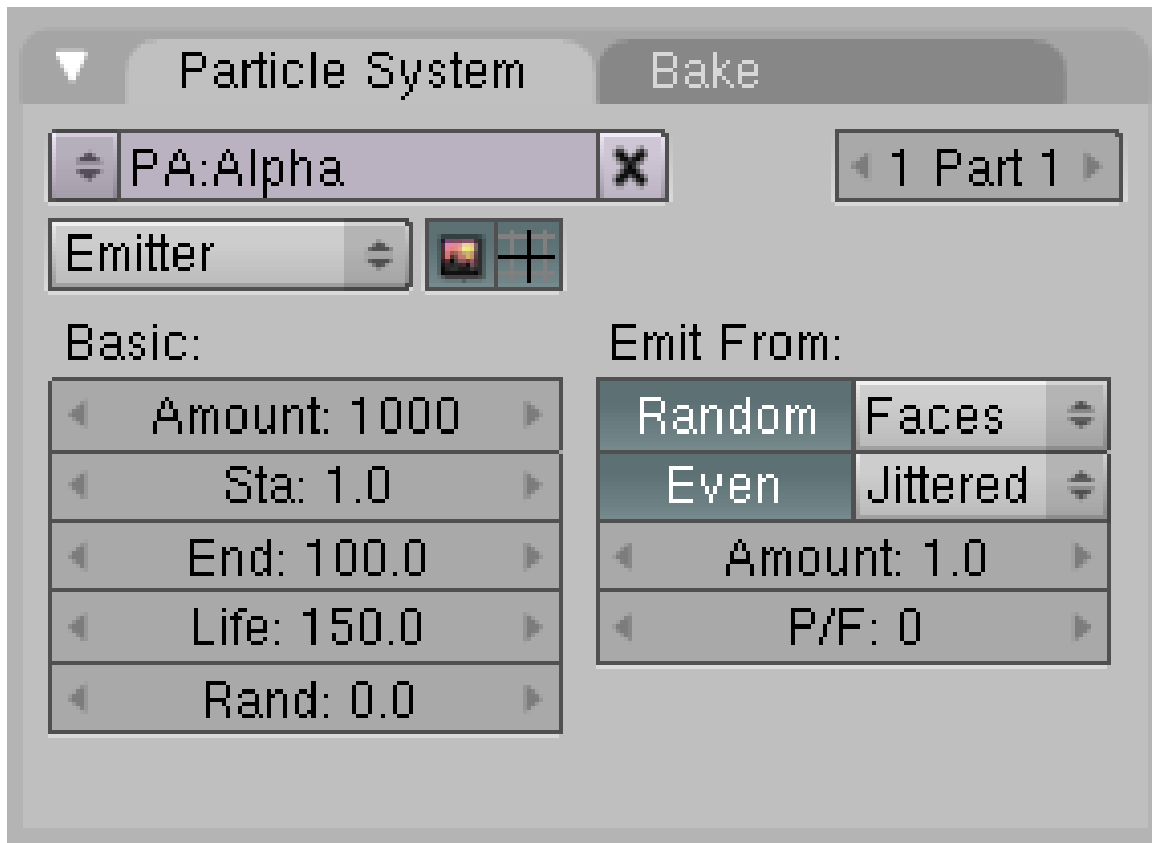


Figure 1091 **Figure 2b:** particle system of the cube.

- Insert a cube.
- Scale the cube as small as the track of the laser beam (the welder) shall become.
- Select the cube, than also select the curve object with **Shift-RMB**.
- **Ctrl-P->Path Constraint** creates a *Follow Path* constraint.
- Select only the cube and activate the option *CurveFollow* in the *Constraints* panel in the *Object* buttons.

If you press **Alt-A** now and let run the animation, the cube will follow the shape of the curve object, but not necessarily in the desired direction and not from the desired location. So some work needs to be done:

- Change to the *Edit* mode of the curve object.

- *Curve->Toggle Cyclic* opens the curve object, by adding (*Subdivide*) of one or more control points you achieve the desired movement. **W->Switch Direction** reverses the direction of the motion of the cube.

By adding a *Time* Ipo-curve to the curve you could set the duration and speed of the animation, but I've left the preset duration of 100 Frames. The last 50 Frames the path will glow a bit and the smoke will vanish.

- Add a particle system to the cube, call this *Alpha*.
- Set the particle lifetime to 150 (or higher) (**Fig. 2b**). The particles will live the entire length of the animation.
- *Bake* the particle animation.
- The cube gets a material:
 - *Halo*
 - RGB 1/1/1, that is completely white
 - *XAlpha*
 - *HaloSize*: 0.03
 - *Hardness*: 40

I want to achieve with these settings, that we will get a grayscale image, that is completely white where the material shall become transparent. We don't need partial transparency, since a relatively sharp cutting edge shall be created. Slight irregularities in the periphery are wanted.



Figure 1092 Figure 2c: A frame of the AlphaMap.

To render the animation, the camera is centered directly over the object.

- *View->Top*
- **Strg-Alt-Num0** sets the camera to the current view. Select the camera.
- With the *Transform Properties* panel in the 3D-window (*Object->Transform Properties*) set the coordinates of the camera exactly vertically above the center of the 3D window ($X/Y=0$).
- In the *Editing* buttons of the camera set the option *Orthographic* in the *Camera* panel. Scale the size of the animation with the *Scale* value.
- The the world color to black in the *World* buttons.
- Render the animation.

Blender now generates an image sequence from 1 to 150 in the subfolder *AlphaMap* (**Fig. 2c**).

176.2. Welding sparks

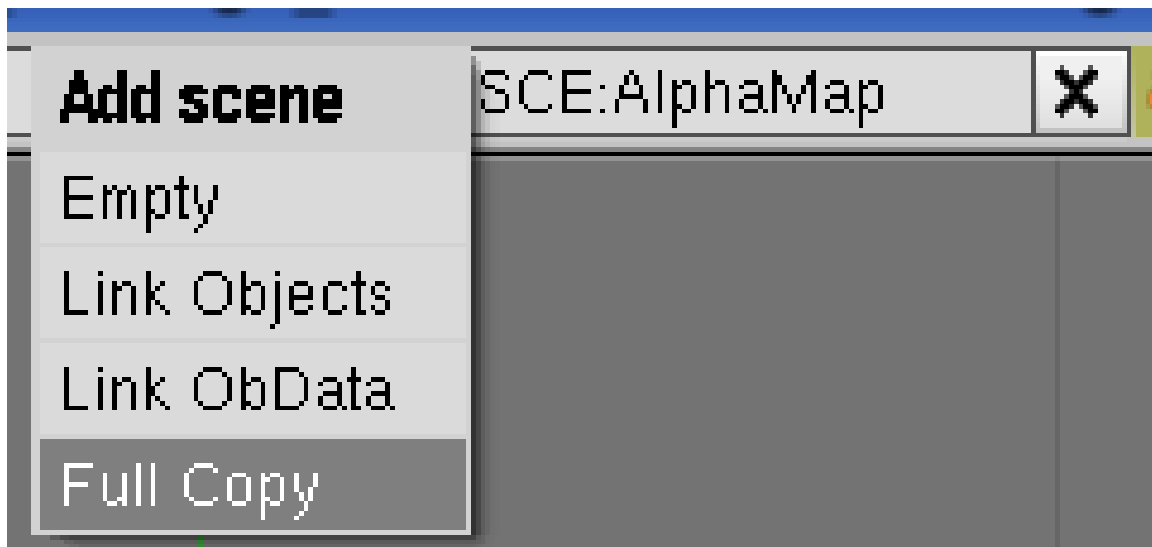


Figure 1093 Figure 3a: Adding a new *Scene*

Now we'll create a new scene, in which the different effects are joined.

- Rename your current scene to *AlphaMap*.
- Create a new scene as *FullCopy* (**Img. 3a**). Name this new scene *Cutter*.

The welding sparks are created using a particle system.

- Add a *UVSphere* with 16 segments and 16 rings.
- Delete the lower half of the sphere in Edit Mode and scale it to approximately the same size as the cube.
- Switch to Object Mode.
- Select the cube in addition to the sphere. Copy the *Follow-Path-Constraint* from the cube to the sphere.

Note that you copy attributes from the active to the selected objects. The active object is the one you've selected last

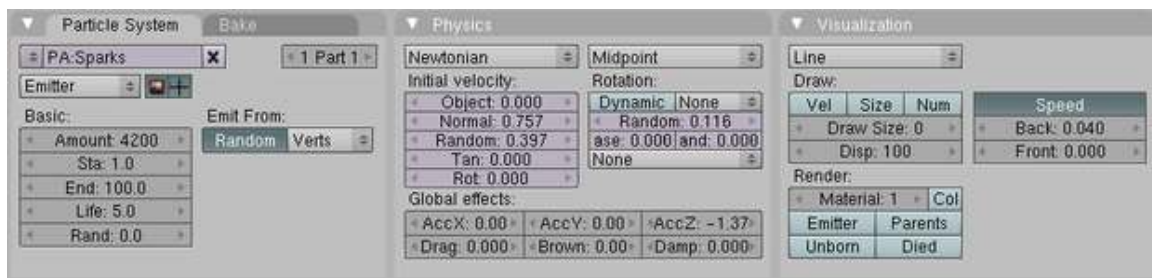


Figure 1094 Figure 3b: The particle system for the sparks.

- Create a particle system for the sphere.
 - Name *Sparks*

- *Amount*: 4200
- *Life*: 5
- *Emit From*: *Random/Verts*
- *Normal*: 0.757
- *Random*: 0.397
- *Rotation Random*: 0.116
- *AccZ*: -1.37

Except for the lifetime the values are relatively arbitrary, I have just experimented with the values until I was halfway satisfied with the results. The sparks have a slight acceleration in negative Z direction to sink down.

- *Visualization*:
 - *Line* (which corresponds as much as possible to the setting *Vect* in the particle system before 2.46)
 - *Speed*
 - *Back*: 0.04
 - *Front*: 0.0

You achieve a long spark with these settings.

- *Bake* the particles. Change to the *Bake* panel and click on the *Bake* button. The particle animation is now permanently saved.

The *Bake* end value is also the final value for the normal particle animation, even if you don't bake. If you need more than 250 frames, you need to increase this value.

176.3. Using the alpha map on an object

The alpha map is now used as texture on a plane. Therefore we have to position the texture exactly like the previously rendered animation, so that the sparks run in sync with the blending out of the plane.

- Add a new plane to your scene (*Add->Mesh->Plane*).
- Scale the plane at about twice it's size - as big as you need it for the final shot.

The main problem is now to fit the texture exactly to the plane. We will give the plane UV coordinates, than fit the plane to the ratio of the texture and finally adjust the position in the 3D window with UV coordinates.



Figure 1095 **Figure 4a:** Setting the *Draw Type*

- Switch to *Draw Type Textured* (**Fig. 4a**).
- Split the 3D window, and open in one part the *UV/Image Editor*.
- Switch to edit mode of the plane.
- Unwrap the plane ((*Mesh->UV Unwrap*)).
- Load an image of the alpha map in the image editor (not the first one, so you can see the course of the map).

Normally the texture is not orientated the way I would like it to be.

- Click the padlock icon in the toolbar of the *UV/Image Editor* (*Update other windows in real time*).
- Rotate the UV vertices in the *UV/Image Editor* until the texture fits (e.g., while holding the **Ctrl** key)
- Switch to object mode.



Figure 1096 Figure 4b: Material settings for the plane

- Add a new material to the plane and name it *Rust*.
- Select the fifth texture slot and add a new texture. Name it *AlphaMap* with the parameters:
 - *Map Input: UV*
 - *Map To:*
 - *Alpha*
 - *Spec*
 - *Nor*
 - *NoRGB*
 - *DVar: 0*

With these (**Fig. 4b**) settings we achieve, that the white pixels in the texture set the Alpha and the Spec value to zero (*DVar*). Because the texture delivers RGB colors we have to activate *NoRGB*, because we need intensity values. We don't need *RayTransp*, because the world background is black anyway. We can't use *ZTransp*, because Halos are not rendered in front of materials with *ZTransp* (probably a bug in Version 2.48a). With *Nor* we get a 3D effect.

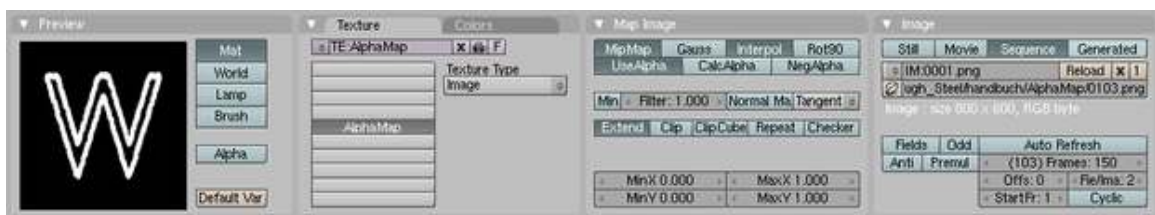


Figure 1097 Figure 4c: Texture settings for the animated alpha map

- Switch to the *Texture* buttons.
- Type: Image Texture
- Load the first image of the alpha map sequence.
 - *Image: Sequence*
 - *Frames: 150*
 - *Map Image: Extend (Fig. 4c)*

Now we adjust the UV coordinates.

- *Object->Transform->Scale to Image Aspect Ratio*. This scales the ratio of the plane automatically to the aspect ratio of the image texture.
- Change to a frame where you can judge the particle animation and it's fitting to the texture.
- Switch to edit mode. Scale the UV coordinates in the UV editor until the texture fits to the particles.
- Switch back to object mode.

176.4. Material for the plane



Figure 1098 **Figure 5:** Material for the plane

We will now give the plane a better material. To judge the material we will first re-focus the camera, give it a texture and make some material settings. If you already have a good metal material, you can use this of course directly and use the Alpha texture from above on your existing material.

- Select the camera.
- Turn *Orthographic* in the *Camera* buttons off.
- Position the camera as desired.
- Select the plane.
 - I've used the rust texture that is linked at the bottom of the page as image texture.
 - *Map Input: Orco*, *SizeX/Y* each 0.5. The size change is to use a larger part of the texture on the final image. But this depends on the size of the camera and the texture.
 - *Map To: Col/Nor*
 - In the second texture slot I've mapped a *Stucci* texture to *Spec*.
 - This abridged sentence means: select the texture slot. Set *Map To* to *Spec*. Deselect *Col*. Change to the *Texture* buttons. Select *Texture type Stucci*.
 - The *Specularity* is thus determined by a texture. The spec value is increased by the texture (*DVar=1*). Therefore you have to set the *Specularity* in the *Shaders* panel to the smallest value the material shall have (e.g. 0.08).

If you have done a test render, you'll see that the particle track of the cube is still there.

- We don't need the cube anymore, you may delete it. In the *Scene Cutter* we use the already generated AlphaMap.

Texture with OSA (*Antialiasing*) are somewhat blurred, for our purpose we use *CatRom* instead of the default *Gauss* method.

- Set in the *Render* panel *CatRom* as *AntiAliasing* filter.

We need a new output directory for the animation, to not overwrite the AlphaMap we've created earlier.

- Set the output directory to `"//Render/"` in the *Output* panel.

176.5. Lighting the torch

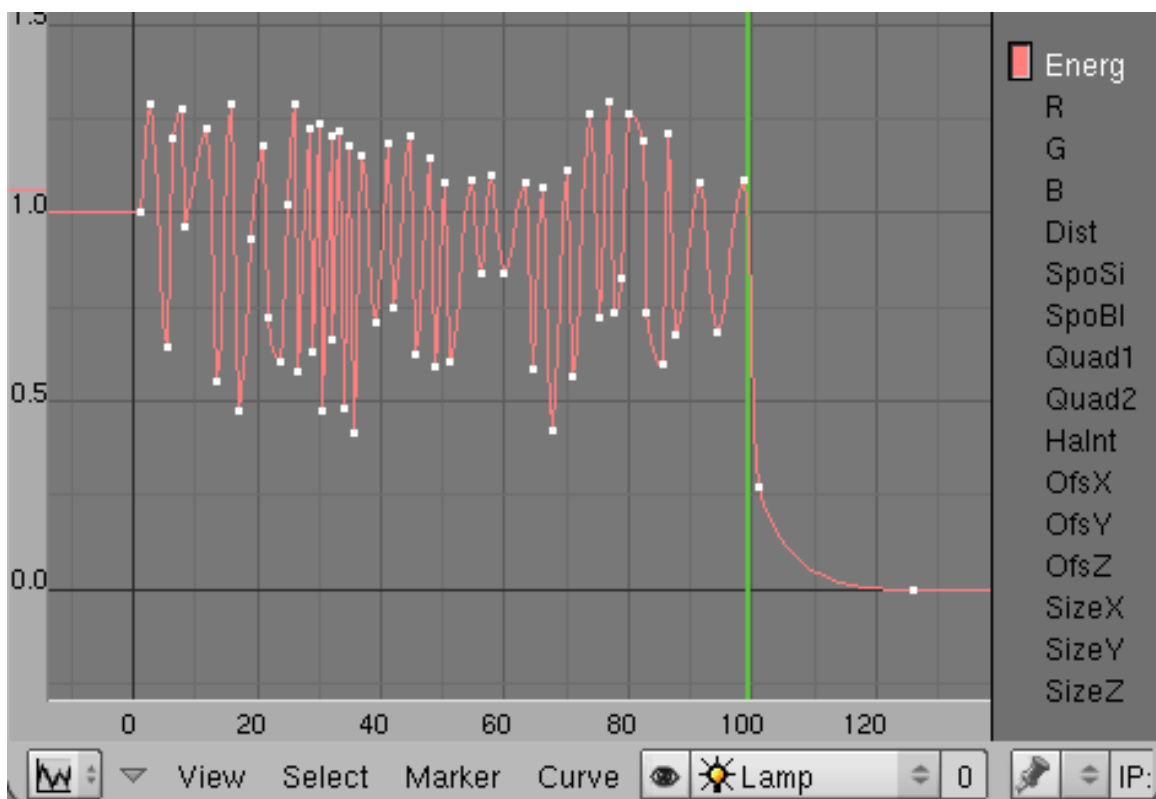


Figure 1099 **Figure 6:** Ipo curve for the energy of the lamp

The torch itself shall cast light, therefore we parent a *Lamp* to the *Sphere* and animate the energy of the lamp.

- Change to frame 1.
- Select the *Sphere*.
- Place the 3D cursor on the *Sphere*. *Object->Snap->Cursor to Selection*
- *Add->Lamp->Lamp*

- Select the lamp in addition to the *Sphere*.
- *Object->Parent->Make Parent->OK*
- Select the lamp and move it a bit up (i.e. in positive z-direction). The lighting shall emit from the sparks, not from the hole.

I find it cumbersome to animate the energy of the lamp by setting ipo-keys in the lamp buttons. Instead I do it directly in the 3D window.

- Change to the *Ipo Curve Editor* in one of the open windows.
- Change the ipo type to *Lamp*.
- Activate the *Energy* channel with **LMB** click in the ipo window.
- By holding **Ctrl** key pressed, you create by left-clicking in the ipo window the points of the ipo curve.

These values may be larger than 1. However, the last maximum value should be reached in frame 100, and decrease significantly to frame 105. The sparks are only issued up to frame 100, and live up to the 105th frame. After that the metal is glowing a bit.

Your result may look somewhat like mine in **Fig. 6**. If the amplitude of the curve is not large enough, you can scale the ipo curve at the y-axis.

176.6. More realistic sparks



Figure 1100 **Figure 7:** Better sparks

You may animate the material for *Line* visualization during the lifetime of each individual particle if you use *Halo* for the material, so you could change the sparks color from white to red.

We will show this in the next step, but confine ourself here to make a better, not animated material (**Fig. 7**).

- Add a material to the *Sphere*, name it *Sparks*.
- Activate *Halo* in the *Links and Pipelines* panel.
- Set the RGB color for *Halo* to white.
- Activate *Lines/X Alpha/Shaded* in the *Shaders* panel.
- *Halo Size*: 0.1
- *Add*: 0.8

176.7. Smoke



Figure 1101 Figure 8a: Sparks and Smoke

Now we're going to create an individual material animation for every particle during its own lifetime.

We will use particles with a simple texture and animate the alpha value of the material. We're going to duplicate the *Sphere* and give it a new particle system.

- Duplicate the *Sphere* with *Object->Duplicate*. The new sphere already bears the *Path* constraint.

-
- Remove the *Sparks* system from the new *Sphere*.
 - Create a new particle system. Name it *Smoke*.
 - *Amount*: 300
 - *Emit from*: *Random/Verts*
 - *Initial velocity*:
 - *Object*: 0.02
 - *Normal*: 0.1
 - *Random*: 0.01
 - *AccZ*: 0.05
 - Bake this particle system.

 - Change to the *Material* buttons.
 - Create a new material and call it *Smoke*.
 - *Alpha*: 0.034. This is the basic visibility for the smoke texture.
 - Add an ipo key for *Alpha* in frame 1.
 - Change to frame 100.
 - Set *Alpha* to 0.0 and add the second ipo key.

Now the smoke will disappear within the lifetime of each particle, independently of the lifetime of the particle.

- *Halo*
 - *Halo Size*: 0.224
 - *Hard*: 12
 - *Add*: 0.014
 - *Halo Tex* This is the crucial setting to use a texture for the shape of the halo.
 - *X Alpha*

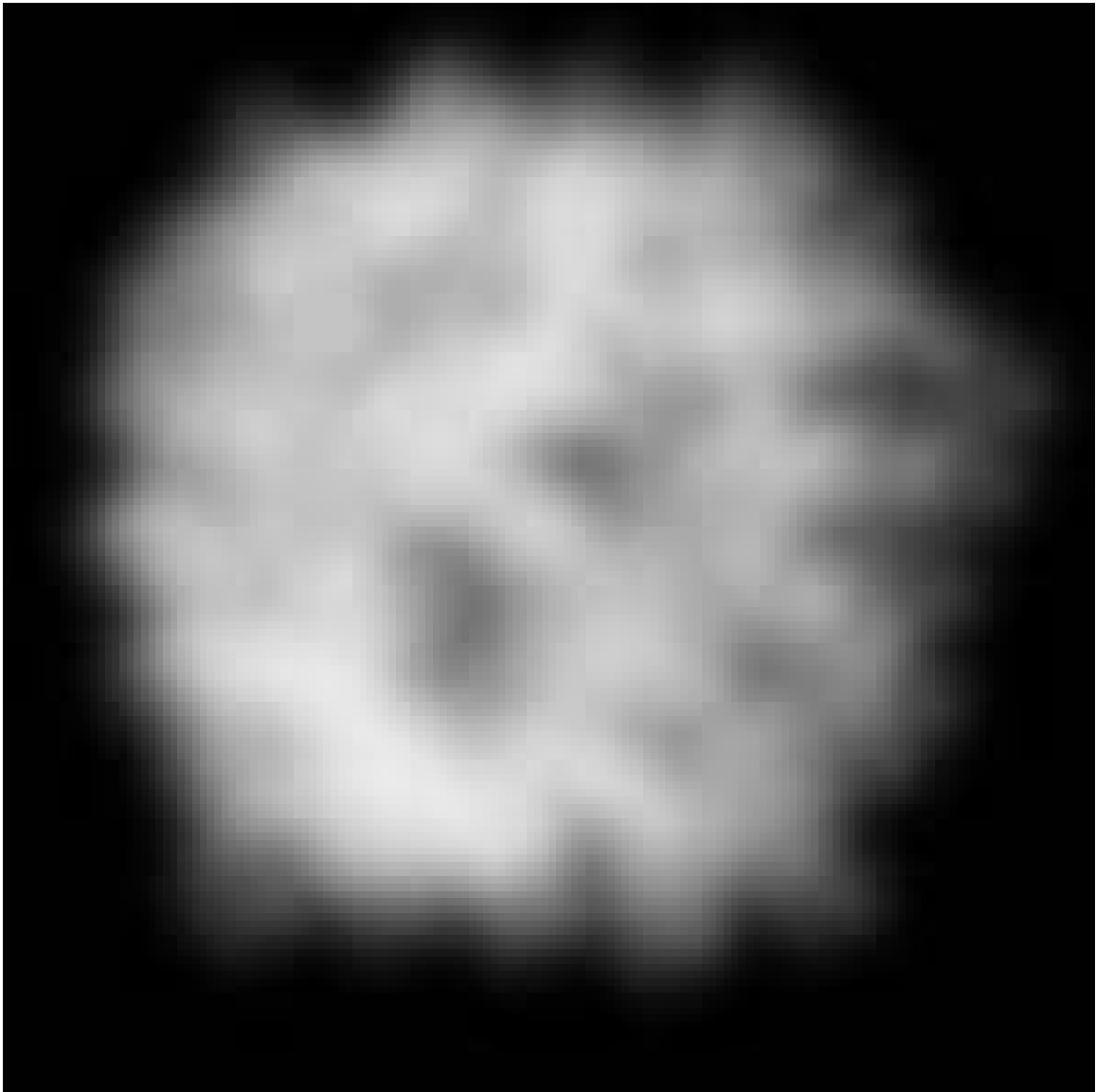


Figure 1102 **Figure 8b:** Particle texture

- Create a new texture, you may also call it *Smoke*.
- *Texture Type: Image*
- Load an image with random, smoke-like structures, like that from **Fig. 8b**.

If the fading goes to quickly, or the texture is to transparent / opaque, change the animation of the *Alpha* value.

You see, even though we use only a few objects and materials, it is extremely helpful if we designate meaningful names to the objects, materials, textures etc.

176.8. After Glow

Now at last we will create another animated texture, that shall simulate the annealing of the steel plate after it is being cut through.



Figure 1103 Figure 9a: A Frame of the *Glow-Map*

- Change to the scene *AlphaMap*.
- Add a new scene again as a *Full Copy*.
- Name this scene *Glow*.
- Enter the output directory `"//Glow/"` in the *Output-Panel* of the *Render* buttons.

The texture shall have nice semi transparencies.

- Therefore we select a different output format in the *Format* panel, namely *RGBA* (RGB with Alpha).
- Selecting *Premul* in the *Render* panel the world background will become fully transparent, and the particle trace semi transparent.

We use the existing particle system, but change it's material settings.

- Call the material of the cube *Glow*.
- Increase the *HaloSize* to 0.04. Our new texture has to be larger than the cutout of the plane, else it will not be visible.

Now we're going to animate the material of the halos. Again this works only because we have *Point* visualization of the particles.

- Change to frame 1.
- Set an IPO key for the RGB color (here white).
- Go to 3rd Frame.
- Change the Halo color to 1/1/0.42 (light yellow). Set the next IPO key.
- Go to 6th Frame. Change the Halo color to 1/0/0 (red). Set the next IPO key.
- Go to 11th Frame. Change the Halo color to 0/0/0 (black). Set the next IPO key.

The particle trace will fade out during 11 frames.

- Render the animation.



Figure 1104 Figure 9b: A finished frame of the animation.

- Go back to the scene *Cutter*.
- Select the plane.
- Load the newly created texture in the third texture slot as an animated image texture.
 - *Add New*. Name the texture *Glow*.
 - *Map Input: UV*
 - *Map To: Col/Emit*

We need the alpha values in the image for changing the emit value. Also the alpha blends the glow texture nicely with the plane texture.

- *Texture Type: Image*
 - Select the first image from the subdirectory *Glow*.
 - Activate *Sequence*.
 - *Frames: 150*
 - *Extend*

The settings for the texture are identical with those of the texture *AlphaMap* (**Fig. 4c**), *except of course for the directory in which the images reside*.

In **Fig. 9b**, see a finished frame of the animation. For the video in **Fig. 1** I've changed the animation a bit, scaled the emitter objects a bit smaller, changed the sparks Color and used a different camera perspective.

Suggestions, critics and approval please write on the commentary page of this article. If you find some obvious mistakes you may correct the article directly of course.

176.9. Links



Figure 1105 Rust Texture

- The Blend File²

² <http://www.snapdrive.net/files/180816/CuttingThroughSteel.zip>

177. Advanced Game Engine

This set of tutorials is for Blender users who stumble up on the basics of certain games, such as making a HUD, menu, or even a simple button. This is also a place where users who know cool ways to spice a game up or explain troublesome aspects of making certain parts of a game genre's content work (like how to get Blender to calculate the damage of a bullet shooting).

There are a couple subsections to this. One is for those who don't like the messiness of python. Learning python to the point of being able to create a game with it will take months, maybe years, just to learn it, not to mention how long it would take to type the specific code and troubleshoot. So, the first section is for those who are much more comfortable with using the GUI Blender screen to make the majority of their game, and the tutorials exist to help them enhance their game.

Note: No matter how much GUI you'd like to use in game creation, there is a certain amount of python you will HAVE to use to be able to really make your game good. Make sure to at least read up on it. Sorry.

Jump to *Game Creating Techniques within a GUI*¹

Then there is the second subsection. This section is to help those who are comfortable with python learn how to use the code in ways they never thought of, and to show those who don't do so well with it just what python can do. While it's a simple language, it can still be incredibly complicated at times, and very, very picky.

Note: DO NOT SUBMIT ANY CODE YOU DO NOT WANT COPIED A MILLION TIMES! This section is for helping the broad and the narrow range of Blender users to learn how to use python better, and there will be no expectation of attribution for submitting code you found out yourself.

Jump to *Game Creating Techniques with Python*²

¹ <http://en.wikibooks.org/wiki/%2FGame%20Creating%20Techniques%28GUI%29>

² <http://en.wikibooks.org/wiki/%2FGame%20Creating%20Techniques%28Python%29>

178. Game Creating Techniques Part One: The GUI

178.1. Making Blender Games Graphically

If you're reading this section, I'm guessing that you have either made a game before or at least are in the process of making one right now. Let's face it: It's probably much harder than you imagined it. A large amount of people take video games and the work put into them for granted. And there are probably quite a few times in which you wanted to do something so simple—such as a menu or even a button—but didn't even know where to start or what to do.

This is why this section exists.

The next few pages will all be tutorials with different techniques on how to make objects in Blender move and respond to things like your mouse, keyboard (a.k.a. "HotKeys") and making it all look like what a real game does. They will be simple versions of a menu, or a button, in which you can use this and put it into your own idea with your own graphics and your own little tweaks.

There will be a certain amount of Python necessary in the next few tutorials, mainly just to allow the mouse icon to appear on screen. However, you can do the rest of the tutorials in the 3D View Window, and under the Logic Index in the Tab window.

179. Creating Pop-Up Menus

In this tutorial, we will be making a Title Screen and a Main Menu Screen to come from that Title Screen. While most people hardly pay attention to these areas on a real video game, they do make the game seem much better to play, and gives the player a certain choice of what they can do before they play the game. Before beginning this, make sure you familiarize yourself with several logic sensors and actuators, such as "Scene" and "Property", as they will heavily be relied upon. It is also best if you have several Pictures ready to use for your game or are ready for additional functions with certain programs (a Word Processor Program and Image Editing Program [that has a transparency setting] can help move along your game.)

179.1. Making An Easy Title Menu

The Title Screen will be very simple -- it will be the "Press Enter to Continue" kind. This will lead to a more interactive kind of Main Menu, in which you can use either key commands or the mouse pointer. This tutorial will not be the most visually pleasing, but then again, you can always add whatever graphics you would like. Again, it would be wise to have handy a word-processing program and a program like Microsoft Paint. "Print Screen" will be a useful function to use while following this tutorial -- the corresponding button is generally located to the right of the F12 key in the row of function keys on your keyboard.

If you have never used different scenes before, you're about to get a crash course. Also, Blender hotkeys and actions will not be explained with (A-key) or (numpad-6); it is assumed you have already learned at least how to perform basic Blender functions.

179.1.1. Making the Menu

First, start up Blender and delete everything in the scene. Then, move to top orthographic view (press numpad 7, or if you have not a numpad, go to the hidden top menu and in the System & OpenGL section select "emulate numpad").

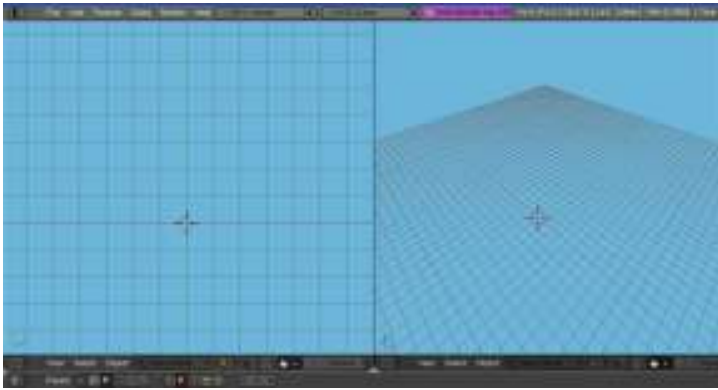


Figure 1106

The first thing you should do is add a new scene. In the Information view panel at the top, click the SCE: drop-down menu and click on ADD NEW. Label this new scene something like "Main Menu". Now go back to your original scene and name it something like "Title Menu". These names will be important when it comes to changing the scenes around. So, on your "Title Menu" Scene, begin by adding a Camera. Move this Camera in the Z direction about a point, just to get it above the grid. Now, set the camera view to Orthographic. Now switch to Camera View. Add a plane that is scaled to cover the entire surface of the Camera's view-bounds. Move that plane just a little bit in the -Z direction. Then, add another plane, but scale this plane to be a thin rectangle about 2/3rds down the Y-length of the Camera, centered in X-direction.

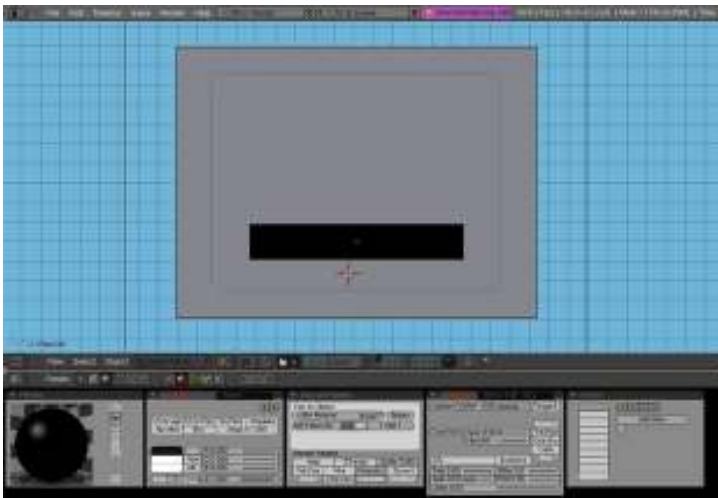


Figure 1107

(Image Shown with Materials for visual purposes, DON'T ADD MATERIALS)

Now this next step isn't necessary, but if you want a good-looking game you will probably need this step.

Let's add some UV textures. Open a good photo/image-editing program like Photoshop, GIMP, or Microsoft Photo Editor; something that preferably supports Transparency. Type some text appropriate for your game such as, "Press any key to continue." What I do is add lots of extra space and add

additional text that I will use in the main menu too, such as "New Game", "Continue", "Story Mode", etc. so it is all in one image for multiple UV textures. If your editor supports it, set the background color to transparent. If not, set it to black so it will register as 0 on the alpha channel in Blender. Save the image (GIF or PNG are reasonable choices for use with transparency). Now, in Blender, split the 3D screen into 2 screens, and change one screen over to UV face select. Select your small rectangle, and switch to Edit Mode and Unwrap the face in the 3D window, then open your saved image in the UV face window. Scale your face in the UV window so it holds only the "Press Any Key to Continue" graphic. If you wish, also add a starting Image on the larger rectangle.

179.1.2. Setting Up The Actions

Now this step is necessary. Select the large background rectangle, in the Buttons Window at the bottom switch it to the Logic Tab. This will be very simple. Add one Sensor, one Actuator, and one Controller. Make the sensor a Keyboard Sensor, depress the "True Pulse" button, and in the key area depress the "All Keys" button. Make the Controller an "AND" controller. Make the Actuator a Scene Actuator. Select the Drop-Down box and make it "Set Scene" wit "SCE:Main Menu". **Before you test the game, make sure to save and add a Camera to the "Main Menu" Scene. Otherwise the game WILL crash.**

If you want the option of returning to this scene, instead of one Actuator make two. Make both of them Scene, but make one "Suspend Scene" with "SCE:Title Menu" and make the other Actuator "Add Overlay Scene" with "SCE:Main Menu". Also, that means in the Main Menu Scene later you'll have to add some key to hit or some action to "Remove Scene" "SCE:Main Menu" and "Resume Scene" "SCE:Title Menu".

179.2. Author's Starting Menu Example

The following contains actual in-progress work by the original Author of this page. The work and ideas are heavily copyrighted and it is highly encouraged (begged) that you do not rip this content off--the author has submitted this work generously to help less advanced users more efficiently.

The following is my work on a game I am making, and I hope it helps you too.



Figure 1108

This will get you to the next menu, which will be on the next section.

179.3. A Main Menu

This menu will be more difficult, using a mouse-python function to use and make the mouse appear when you play your game and a maze of sensors and actuators entangled.

Here is the first thing you need: a python script to make the Mouse show ---

**This code is attributed to the author of the chapter "Blender Game Engine/A Simple Mouse Pointer" and this is not my own code.*

```
import Rasterizer as r
r.showMouse(1)
```

Put this in the text editor, and save the text as something simple like "Mouse". Now, on your Main Menu scene, select the camera and add 1 sensor, 2 controllers, 1 actuator, and 1 property. Name the property "Switch" "Int" Type, and set it to 0. Set up your logic as follows:



Figure 1109

A change will need to be made in order to make yours work. The python controller should be Script:Mouse, not Script:showpointer. If you can't see the connections:

- Connect the Sensor to the 2 controllers
- Connect the AND controller to the actuator
- Leave the Python controller open-ended

If you play the game, you will see the mouse then! Now, back to the Menu, there are multiple kinds of Menus you can make when it comes to Menus popping up. The one I believe works best is a 3-layered button. To explain what this means, I'll run you through the steps of making one. Let's start with the layers themselves. (For this to work best, make the background of any images you use have transparency, so you can make invisible-yet-clickable faces.) Create a rectangle facing the camera where you want your first menu option to be. Copy the object until you have 3 rectangles, and make each one very slightly in front of the last. Now, Select the back rectangle, and on this one apply the UV face of the option's words. The second rectangle is the UV face for a graphic outline around the words if the option is selected in the game. It's not necessary but then again, it helps the player with selecting options and it looks nice. The top rectangle will be the selectable one--make the UV face applied only over a completely transparent area. Now, for the logic.

Text-Rectangle:

-No Logic Needed

Outline-Rectangle:

-Give it a Property named "Selected", Int Type, starting at 0.

-Make a Property Sensor, True Pulse, "Selected" Property is Equal to 1, Connected to an AND controller, Connected to a Visibility Actuator at "Visible"

-Make a Property Sensor, True Pulse, "Selected" Property is Equal to 0, Connected to an AND controller, Connected to a Visibility Actuator at "Invisible"

Creating Pop-Up Menu

-Make a Property Actuator, "Selected" Property is Assigned Amount 1
-Make a Property Actuator, "Selected" Property is Assigned Amount 0
-Create AND Controller and connect both Mouse Sensors. Make this AND Controller connect to the Outline-Rectangle's Property Actuator "Selected" Assign=1.

Transparent-Rectangle:

-2 Mouse sensors: "Mouse Over" and "Left Button". Set them to "True Pulse".

(Click Image to Enlarge)

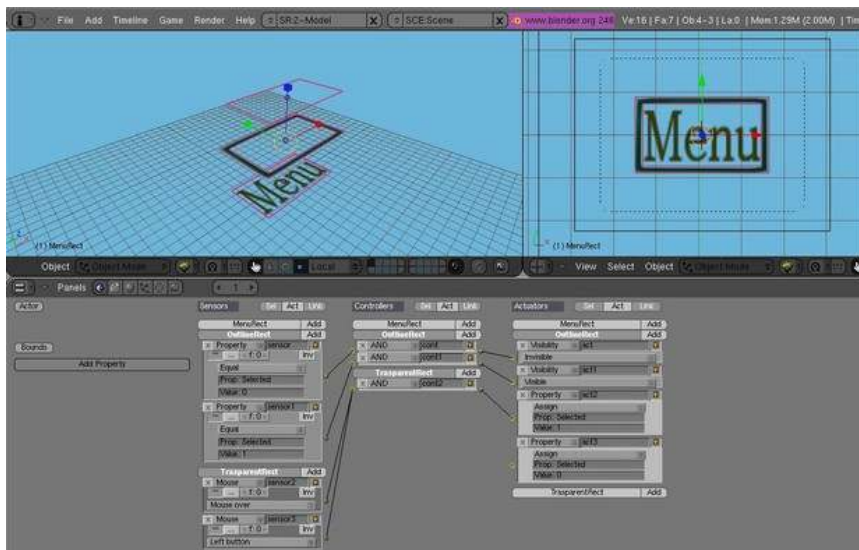


Figure 1110

Now here is the trickier part to explain. To make things a little more simple, name that last Outline-Rectangle's AND controller connected to the 2 mouse sensors. Select all 3 layers, and duplicate them 3 times. Make each one (in camera view) below the last, making sure to keep them parallel on the Z axis. Now, to make only one selected at a time, take the named controller of the first, and connect it to the Outline-Rectangle's Property Actuator "Selected" Assign=0 of second and third button. Do the same from the second button to the first and third one, and same with the third to the first and second. It should look like the picture:

Click Image to Enlarge

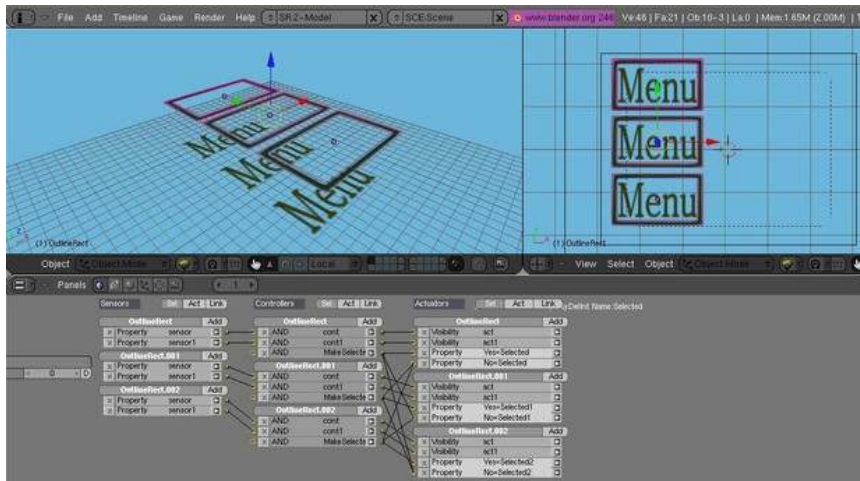


Figure 1111

This logic arrangement should yield (when the game is played) options in which if you click one it will light up, and if you click another, that one will go down and the other will light up.

Now this looks nice aesthetically, but the use of it is now all you have to do is make some Scene Actuators. Add 2 more actuators to each of the Transparent Planes. Use the same method as you did with the Properties, but instead of "Property Assign 1/0" make 1 AND controller, 1 Scene Overlay and 1 Scene Remove for each transparent button. Now make 3 additional Scenes, and either make an entirely new menu for each, or to be easier just make colored rectangles on the scene, different colors for each scene -- either way, make sure that, if the scenes overlay, the additional scene won't cover up the buttons of the Main Menu Scene. So, now connect the 2 Mouse Sensors to the corresponding AND Controller, and connect that to the Corresponding "Overlay Scene" Actuator and 2 Opposing "Remove Scene" Actuators. Do this with the other 2 Transparent Rectangles. Make sure your new scenes also have cameras and that they are scaled properly.

Click Image to Enlarge

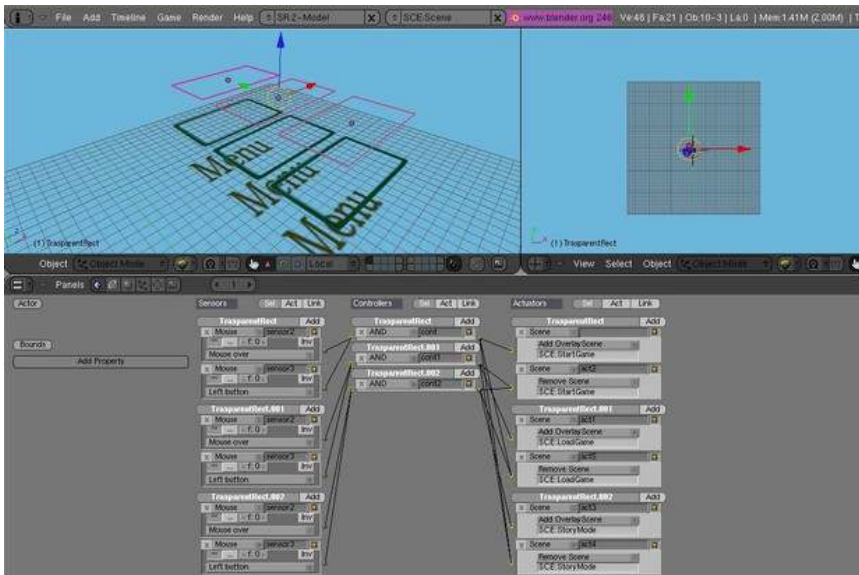


Figure 1112

You can use this technique to make Menus and Sub-Menus Galore! It's fairly simple to make multiple sublevels of this and get back to the original scene with this method.

I hope this helped you make your menus work well.

180. Creating Moving Menus

In this section, you are going to learn a variety of ways to make menus with motions. These menus can be used in more ways than the simple pop-Up menus, such as in-game if you need to select something you can use this to bring up a menu smoothly. In this section there is heavy use of Properties, "Property" Logic Blocks, and especially IPO usage. When you do it once, it will be very much easier to do it again.

180.1. Making Moving Selectors

To kick this tutorial off, let's just do a simple icon that will be next to the selected menu item. There will be 6 possible menu items, which can be controlled with both the arrow keys and the mouse. Let's begin by booting up Blender, and deleting everything--then adding a camera.

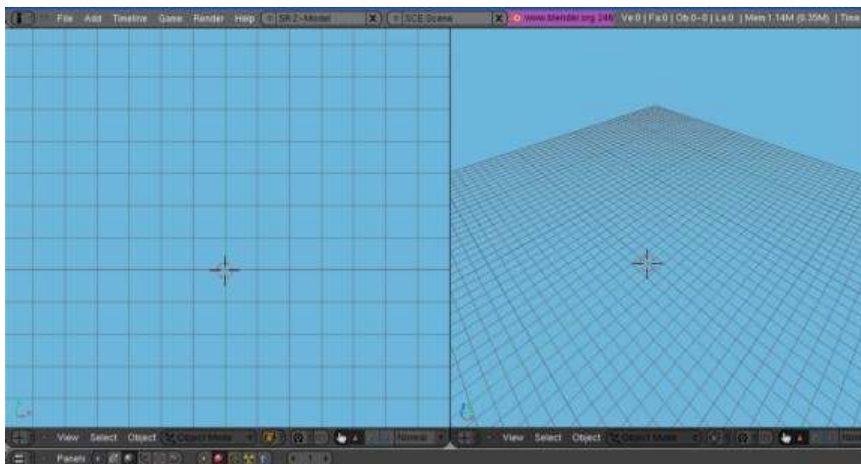


Figure 1113

Set this camera to orthographic. Now I like making things look nice, but here are the basics of what you will need for this tutorial:

- Make 6 Planes, all different objects, and put them in a vertical list.
- Make a circle next to the first plane. This will act as the Selector.

So after you've made you objects, it should look similar to this:

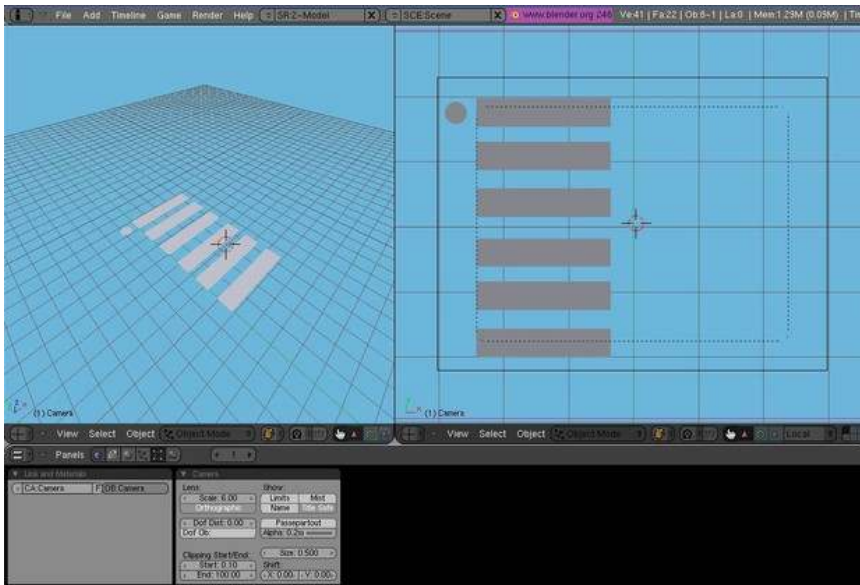


Figure 1114

For me, I made mine look more like this, a little more detailed:

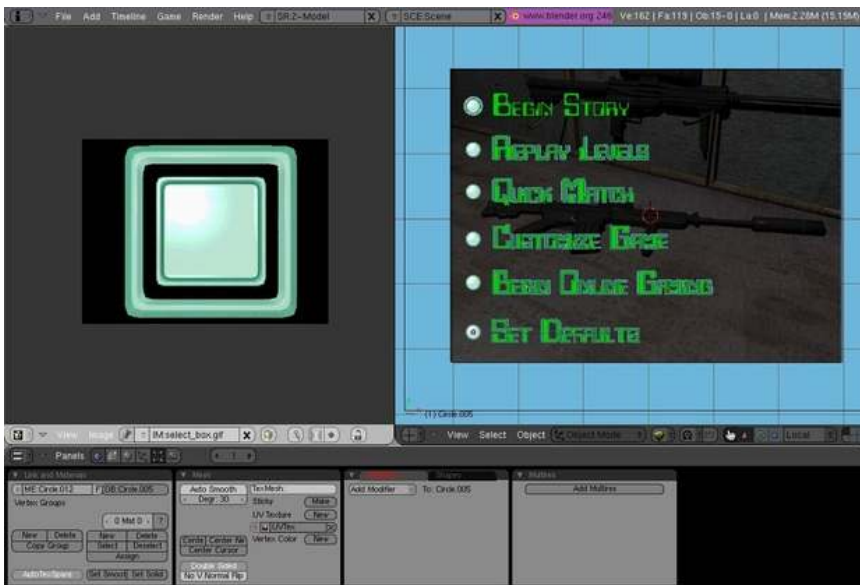


Figure 1115

So now that we have our menu set up, it's time to set up the properties that will tell the selector how to move. Select the selector that will be moving and give it a property name "Selection". Make it "Int" type, and start it out at "1". To get this selector to move the way we want it, we have to set up its movements. Make sure you are on frame 1, and put the cursor on the first menu item, then Insert Keyframe for "Loc". Next, move to frame 2, and put the cursor on the second menu item, then insert "Loc" Keyframe, and continue in this fashion for all your menu items. To get this applied in the Game Engine, in the logic bricks, add an "Always" Sensor (True Pulse Enabled), connected to an

"AND" connector, and connected to a "IPO" Actuator, set to "Property" with Prop:Selection. Now we can do 3 things with this selector to have it move like a selector in a video game:

Up/Down Arrow Keys

To make the up and down arrow keys work properly, Select the Selector and add four sensors, controllers, and actuators. Make two sensors keyboard, with one set to "Uparrow" and one set to "Downarrow" (do not activate pulse triggering). Make the other two sensors Property Sensors, with True Pulse. For the first, make it "Equal", Prop:Selection, Value:0. For the other, make it "Equal", Prop:Selection, Value:7. For the Actuators, make all four "Property" sensors. For two of them, make them "Add", Prop:Selection, and make the first "Value:-1" and the second "Value:1". The other two should be "Assign", Prop:Selection, with the first at "Value:1" and the next as "Value:6". Now connect them as follows:

Click To Enlarge Image

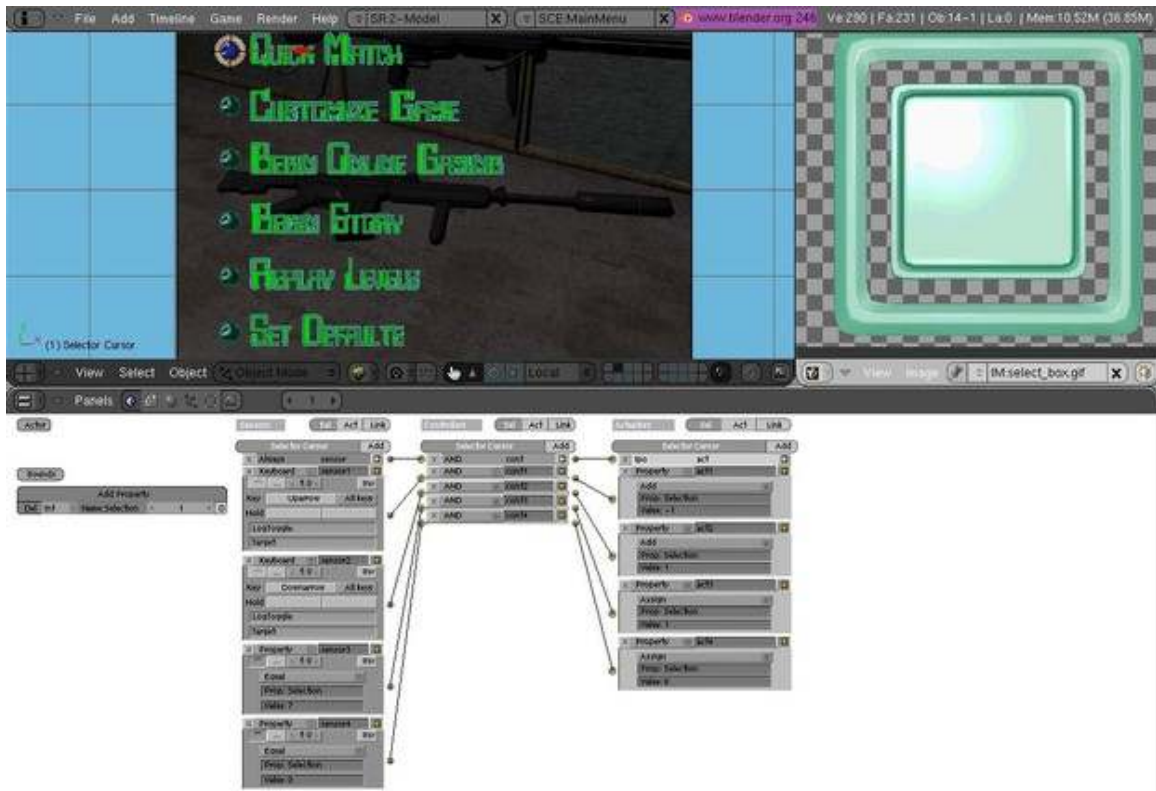


Figure 1116

"Keyboard:Uparrow"--"AND"--"Prop:Selection/Add/Value:-1"

"Keyboard:Downarrow"--"AND"--"Prop:Selection/Add/Value:1"

"Prop:Selection/Equal/7"--"AND"--"Prop:Selection/Assign/Value:1"

"Prop:Selection/Equal/0"--"AND"--"Prop:Selection/Assign/Value:6"

Now you can play your game, and the arrow keys should work just fine!

HotKeys

If you don't know what a HotKey, it's a way to access menu items or action by pressing a specified letter, instead of just clicking or arrow keys. This method will require a different step when you make your UV text textures, as you will need to somehow change one letter (either the color or possibly underline) to let people know that that is a HotKey action.

The HotKey system is easy to set up right alongside the IPO/Selection property. In each menu item, highlight or make special one unique letter in each menu item. Then, with the selector selected, set up 6 keyboard logic sensors, each one for a different letter. Add 6 controllers, and 6 actuators as well. Make all the actuators "Property", "Assign, Prop:Selection, starting with Value:1 progressing to Value:6"

Mouse Selection

Our final selection method we can add is the mouse scroll over. Set up the Mouse Sensor to Scroll Over for all 6 invisible selection planes. Then, make the AND Controllers and connect to your original set of "Set Property" Actuators. that matches the Object's IPO property number. It's fairly simple.

Make the Selection

The final thing to do is, once the selections are set up, make another 8 sensors, 7 controllers, 6 and actuators. Now make the 7th one a keyboard sensor for the ENTER key, as most people usually think they can start a game by pressing ENTER. The 8th Sensor is for another Mouse Sensor for "Left Click" so that if any of them is highlighted and you Left-Click the mouse, then it will be equivalent to pressing the ENTER key. That may be confusing as, unless you set up the dozen extra sensors, controllers, and actuators, you don't need to scroll over the selection to pick it, so that 8th sensor is your choice whether you want to add it. Connect the 7th and 8th sensor to the 7th controller. If you have the 8th sensor, make it an OR controller; otherwise, make the controller AND. As for the first 6 sensors, they should be Property Sensors for "Prop:Selection" "Value:[1-6]". Connect each to their own separate controller. Now connect the 7th controller to all 6 Actuators, and the first 6 controllers connect to their own Actuator alongside the 7th. The Actuators can be whatever you want them to: whatever the selection causes to happen when you select the item selection.

180.2. Make the Menu Move

So we created a selection-based moving icon on a menu, but what might happen if you select that item and need a sub-menu in-game? You can't very well pause the game every time (unless it's a pause menu). This is where making the menu itself move comes in very handy. In this section, you will make an object clickable, and once clicked it will give out a submenu, which in itself will be able to give out a submenu.

The IPOS

181. The "5-Layer" Button

181.1. Scheduled Construction

This chapter is scheduled to be created by the Author. Sorry for the inconvenience.

182. Creating Object Outlines

182.1. Scheduled Construction

This chapter is scheduled to be created by the Author. Sorry for the inconvenience.

183. Game Creating Techniques Part Two: Python

183.1. Needs Expert

This chapter is not yet created, but it is something that would be very helpful to know about Blender. This book would benefit very much so if readers were to be so generous as to give helpful information on this subject, and expand it further. Thank You.

183.2. Additional Resources

- [Blender 3D: Blending Into Python](#)¹

183.3. External Links

- <http://www.theboomshelter.com/hss.html>
- <http://www.youtube.com/watch?v=7hbCEyvVYHI>

¹ <http://en.wikibooks.org/wiki/Blender%203D%3A%20Blending%20Into%20Python>

184. Really Advanced: The Game Engine Source

Mission Statement: The purpose of this section is to show a programmer how to make sense of the blender game engine source code. It is not intended for those who cannot code in C++. If you're one of the people who have a basic understanding, I'm being careful to use keywords that if you google, should provide enough information to understand.

When you press P, several things are sent to the GE before anything can be done. What might these be?

```
StartKetsjiShell(struct ScrArea *area,
                char* scenename,
                struct Main* maggiel,
                struct SpaceIpo *sipo,
                int always_use_expand_framing)
```

Blender uses a lot of structs and classes. You need to have a working understanding of these as well as inheritance before we can move on. For a refresher: http://www-numi.fnal.gov/offline_software/srt_public_context/WebDocs/Companion/cxx_crib/inheritance.html

The SCA_IObject class Blender Trunk\source\gameengine\GameLogic\SCA_IObject.cpp and SCA_IObject.h

This class encapsulates a lot of other things, so we can start here and move downwards to the things that it controls. First off it uses vectors to store all the different Sensors, Controllers and Actuators that are linked to the object. *A vector is like an array, but it is dynamic and has built in functions.* We will begin by examining how sensors are added.

```
typedef std::vector<SCA_ISensor *> SCA_SensorList;
```

```
SCA_SensorList m_sensors;
```

The first line here is a typedef so it just tells us that we can create a vector of SCA_ISensor by using the name SCA_Sensor list.

The second line creates the vector with the name, m_sensors

```
void SCA_IObject::AddSensor(SCA_ISensor* act)<br />
{ m_sensors.push_back(act); }
```

This function allows the parents of an SCA_IObject to add new sensors by calling: SCA_IObject.AddSensor(keyboard); *pseudocode*

So now you know how Objects in the GE create sensors (and actuators and controllers, its the same :) So lets look at how Sensors are grabbed! Since they are dynamically added the program can't make assumptions about how many are there or what they will be. Thus there is this helper function:

```
1'SCA_ISensor* SCA_IObject::FindSensor(const STR_String& sensorname)
2{
3SCA_ISensor* foundsensor = NULL;
4
5for (SCA_SensorList::iterator its =
   m_sensors.begin();!(its==m_sensors.end());its++)
6{
7if ((*its)->GetName() == sensorname)
8{<br />
9foundsensor = (*its);
10break;
11}
12}
13return foundsensor;
14}''
```

Put simply, you give this the name of a sensor and it returns a pointer to the sensor.

for the newbies: I know you learned all about classes when i asked you to, but just in case:) the first word, "SCA_ISensor*" this tells us that the function will return a pointer to a SCA_ISensor. And based on the functions arguments, we know that when you call FindSensor you must give it a string between the parenthesis.

in line 3 you create an empty pointer which you will use to store the sensor that you find. Then you loop and go through each sensor in the vector of sensors (remember, it was named m_sensors) until you find the one that matches the name. If you don't find one, you return NULL.

void SCA_IObject::Suspend(void) What this does is it loops through all of the sensors and sends each one a "suspend" call. What this does is change a variable named, "m_suspended" which makes stops the sensors from running when they are called with their Activate() function.

Part V.
Appendices

185. Glossary

185.1. A

- **Alpha Channel**¹ is an additional channel in a 2D image for transparency. In an image element which stores a color for each pixel, an additional value is stored in the alpha channel containing a value between 0 and 1. A value of 0 means that the pixel does not have any coverage information; i.e. there was no color contribution from any geometry because the geometry did not overlap this pixel. A value of 1 means that the pixel is fully opaque because the geometry completely overlapped the pixel.
- **Ambient Light** is light that doesn't seem to come from a specific source, but is just there. Look under the desk - it's pretty dark, but there's some light there. In the real world, this is caused by stray photons bouncing around and occasionally ricocheting under the desk. Ambient light is the basic, minimal amount of light in the whole scene. Adding too much ambient light makes a scene look washed out. Since the light doesn't come from anywhere, all sides of an object are illuminated equally, and it won't have any shading on it.
- **Ambient Occlusion**² (AO) is a ratio of how much ambient light a surface point would be likely to receive. It simulates a huge dome light surrounding the entire scene. If a surface point is under a foot or table, it will end up much darker than the top of someone's head or the tabletop.
- **Armature**³ is the interconnection of bones that form the skeleton of an animated figure. The Inverse Kinematics library contains the code to make armatures move. The armature must still be rigged⁴ with 3D objects to give shape to its head, hands, trunk, feet, etc.

185.2. B

- **Background image**: a 2D image ("picture") that is placed "behind" the entire 3D scene, like a backdrop on a movie set. Blender permits the placement of these images in all six directions from the origin: back, front, top, bottom, left, right.
- **Bake**: to precompute computationally-intensive elements of an animation. For example, in a physics simulation involving the behaviour of fluids or clothing, you would set up the physical parameters, then compute (*bake*) the positions and shapes of the objects over the duration of the animation. Afterwards, you can assign materials and lighting, and then render the frames to produce the actual animation. Doing the baking as a separate step, and saving the results from that,

1 <http://en.wikipedia.org/wiki/Alpha%20compositing>

2 <http://en.wikipedia.org/wiki/Ambient%20occlusion>

3 http://en.wikipedia.org/wiki/armature_%28sculpture%29

4 <http://en.wikipedia.org/wiki/rigging>

means you can change your mind about the materials and lighting and rerender the frames more quickly.

- **Bézier surfaces**⁵ were first described in 1972 by the French engineer Pierre Bézier who used them to design automobile bodies. Bézier surfaces can be of any degree, but bicubic Bézier surfaces generally provide enough degrees of freedom for most applications.
- **BF** is Blender Foundation
- **Blend** - *to Blend*, working with Blender; also Blender's file extension.
- **Bounce Light**: Simple lighting situations have a single light, called a key light, illuminating one side of an object. This creates strong shading and definition of the volume of the object. However, a 3D light will often make the contrast too great - the dark side of the object is completely black since no light is hitting it. In reality it would still be lit a little, just not as much as the brightly lit side, because of light bouncing around the room and hitting the dark side of the object. In realtime 3D, bounce light is not calculated, so you have to create it yourself. Either add a little ambient color, or put a second, less bright directional light pointing the opposite direction to give a little light to the shadows.
- **Bump mapping**⁶ is a technique where at each pixel, a perturbation to the surface normal of the object being rendered is looked up in a texture map and applied before the illumination calculation is done. Bump Mapping use a gray-scale image map to change the direction of surface normals. You can use this to simulate height, so that you can paint wrinkles and bumps. 50 % grey means neutral (no change is made), lighter means higher, darker means lower. Note that the position of faces is not actually changed; by rotating just the normals, lighting will change too, to give the illusion of a height difference. This has downsides too: the outline of objects isn't changed, so the trick is given away. For similar effects you can use Displacement Mapping and Normal Mapping.

185.3. C

- **Caustics** in optics is a bundle of light rays. For example a caustic effect may be seen when light refracts or reflects through some refractive or reflective material, to create a more focused, stronger light on the final location. Such amplification, especially of sunlight, can burn — hence the name. A common situation when caustics are visible is when some light points on glass. There is a shadow behind the glass, but also there is a stronger light spot. Nowadays, almost every advanced rendering system supports caustics. Some of them even support volumetric caustics. This is accomplished by raytracing the possible paths of the light beam through the glass, accounting for the refraction, reflection, etc.

185.4. D

- **Depth of Field**⁷ (DOF) is the distance in front of and behind the subject which appears to be in focus. For any given lens setting, there is only one distance at which a subject is precisely in focus,

5 <http://en.wikipedia.org/wiki/B%26E9zier%20surface>

6 <http://en.wikipedia.org/wiki/Bump%20mapping>

7 <http://en.wikipedia.org/wiki/Depth%20of%20field>

but focus falls off gradually on either side of that distance, so there is a region in which the blurring is tolerable. This region is greater behind the point of focus than it is in front, as the angle of the light rays change more rapidly; they approach being parallel with increasing distance.

- **Diffuse Light** is even, directed light coming off a surface. For most things, the diffuse light is the main lighting we see. Diffuse light comes from a specific direction or location, and creates shading. Surfaces facing towards the light source will be brighter, while surfaces facing away from the light source will be darker.
- **Directional Light** is a light that has a specific direction, but no location. It seems to come from an infinitely far away source, like the sun. Surfaces facing the light are illuminated more than surfaces facing away, but their location doesn't matter. A Directional Light illuminates all objects in the scene, no matter where they are.
- **Displacement Mapping**⁸ uses a greyscale heightmap, like Bump Mapping, but the image is used to physically move the vertices of the mesh at render time. This is of course only useful if the mesh has large amounts of vertices, but the (relatively) new "Simple Subdiv" subsurf option allows you to add more vertices at render time which will be moved by the displacement. This makes it much slower than Bump Mapping, as there need to be many more faces to render, but it is much more realistic.

185.5. E

- **Environment Maps** (EnvMaps) is the method of calculating reflections. Involved rendering images at strategic positions and applying them as textures to the mirror. Now in most cases obsoleted by Raytracing, which though slower is easier to use and more accurate.

185.6. F

- **Focal Length**⁹ of a lens is the distance along the optical axis from the lens to the focus (or focal point). The inverse of a lens' focal length is called its power.
- **Focus**¹⁰ of a lens is the point onto which collimated light parallel to the axis is focused.
- **Foreshortening**
- **Fresnel**¹¹ lens is a type of lens invented by Augustin-Jean Fresnel. Originally developed for lighthouses, the design enables the construction of lenses of large size and short focal length without the weight and volume of material which would be required in a lens of conventional design.

⁸ <http://en.wikipedia.org/wiki/Displacement%20mapping>

⁹ <http://en.wikipedia.org/wiki/Focal%20length>

¹⁰ <http://en.wikipedia.org/wiki/Focus%20%28optics%29>

¹¹ <http://en.wikipedia.org/wiki/Augustin-Jean%20Fresnel>

185.7. G

- **GE** is Game Engine.
- **Global Illumination**¹² (GI) is a superset of radiosity and ray tracing. The goal is to compute all possible light interactions in a given scene, and thus obtain a truly photorealistic image. All combinations of diffuse and specular reflections and transmissions must be accounted for. Effects such as colour bleeding and caustics must be included in a global illumination simulation.
- **Gouraud shading**¹³ is a method used in computer graphics to simulate the differing effects of light and colour across the surface of an object. In practice, Gouraud shading is used to achieve smooth lighting on low-polygon surfaces without the heavy computational requirements of calculating lighting for each pixel. The technique was first presented by Henri Gouraud in 1971.

185.8. H

- **High Dynamic Range Image**¹⁴ (HDRI) is a set of techniques that allow a far greater dynamic range of exposures than normal digital imaging techniques. The intention is to accurately represent the wide range of intensity levels found in real scenes, ranging from direct sunlight to the deepest shadows. The use of high dynamic range imaging in computer graphics has been popularised by the work of Paul Debevec. Blender uses Yafaray for these techniques.

185.9. I

- **Index Of Refraction**¹⁵ (IOR) is about the way that light passes through different types of materials... diamond, glass, water etc. When a light ray travels through the same volume it follows a straight path. However if it passes from one transparent volume to another, it bends. This is why a straw in water looks bent. The amount of bending differs between materials. The angle by which the ray is bent can be determined by knowing two things: the angle at which the incoming ray has been cast and the Index of Refraction. This IOR value is unique for every material. Glass has an IOR of about 1.5 and water 1.3. By increasing the IOR value for a Blender material, you can control how much the environment behind the transparent object is distorted, and thus improving the realism of the shader.
- **Interpolation**¹⁶ (IPO) is an animation curve: it indicates how the object must "move" between an initial and a final position, at the rate determined by the rendering engine. Objects can be animated in many ways. They can be animated as Objects, changing their position, orientation or size in time; they can be animated by deforming them; that is animating their vertices or control points; or they can be animated via very complex and flexible interaction with a special kind of object: the Armature.

12 <http://en.wikipedia.org/wiki/Global%20illumination>

13 <http://en.wikipedia.org/wiki/Gouraud%20shading>

14 <http://en.wikipedia.org/wiki/High%20dynamic%20range%20imaging>

15 <http://en.wikipedia.org/wiki/Refractive%20index>

16 <http://en.wikipedia.org/wiki/Interpolation>

- **Inverse Kinematics**¹⁷ (IK) is the process of determining the movement of interconnected segments of a body or model, starting from the desired motion of the endpoints of the armature. Using ordinary Kinematics on a hierarchically structured object you can for example move the shoulder of a puppet. The upper and lower arm and hand will automatically follow that movement. IK will allow you to move the hand and let the lower and upper arm go along with the movement. Without IK the hand would come off the model and would move independently in space. The Blender Armature System includes Inverse Kinematics. For general armatures there are many possible solutions to the IK.

185.10. J

- **JPEG**¹⁸ Acronym for Joint Photographic Expert Group (pronounced jay-peg) is a commonly used standard method of lossy compression for photographic images. The file format which employs this compression is commonly also called JPEG; the most common file extensions for this format are .jpeg, .jif, .jpg, .JPG, or .JPE although .jpg is the most common on all platforms.

185.11. K

- **Keyframe** is a frame in an animated sequence of frames that was drawn or otherwise constructed directly by the user. When all frames were drawn by animators, the senior artist would draw these frames, leaving the "in between" frames to an apprentice. Now, the animator creates only the first and last frames of a simple sequence; the computer fills in the gap. This is called tweening.

185.12. L

- **Luminosity**¹⁹ (more properly called luminance) is the density of luminous intensity in a given direction. In astronomy, luminosity is the amount of energy a body radiates per unit time. It is typically expressed in the SI units watts, in the cgs units ergs per second, or in terms of solar luminosities, L_s ; that is, how many times more energy the object radiates than the Sun, whose luminosity is 3.827×10^{26} W.

185.13. M

- **Motion Blur**²⁰ is the simulation of the phenomenon that occurs when we perceive a rapidly moving object. The object appears to be blurred because of our persistence of vision. Doing motion blur makes computer animation appear more realistic. It can be thought of as adding back some of the time dependence expressed in the Rendering Equation.

17 <http://en.wikipedia.org/wiki/Inverse%20kinematics>

18 <http://en.wikipedia.org/wiki/JPEG>

19 <http://en.wikipedia.org/wiki/Luminosity>

20 <http://en.wikipedia.org/wiki/Motion%20blur>

185.14. N

- **Nabla**²¹. *Ton wrote*: Almost all procedural textures in Blender use derivatives for calculating normals for texture mapping (with as exception "Blend" and "Magic). The texture normal, the derivative, is calculated by using four samples in the texture formula:

```
s0= texture(x, y, z)
s1= texture(x+nabla, y, z)
s2= texture(x, y+nabla, z)
s3= texture(x, y, z+nabla)
```

```
normal[0]= s0-s1
normal[1]= s0-s2
normal[2]= s0-s3
```

Up to now, the "nabla" offset was a constant (0.025) which worked fine in most cases, but doesn't give proper control over the way a texture is sampled, for example to make the effect smoother or sharper. This feature especially is useful in combination with the ColorBand feature.

- **Non-Linear Animation** (NLA) allows the animator to edit motions as a whole, not just the individual keys. Nonlinear animation is not just about editing and manipulating groups of keyframes, but it also allows you to combine, mix, and blend motions to create entirely new animations.
- **Nonuniform Rational B-Splines**²² (NURBS) is a computer graphics technique for generating and representing curves and surfaces.
- **Normal** (Surface Normal) to a flat surface is a three-dimensional vector which is perpendicular to that surface. A normal to a non-flat surface at a point p on the surface is a vector which is perpendicular to the tangent plane to that surface at p.
- **Normal Mapping** is similar to Bump Mapping, but instead of the image being a greyscale heightmap, the colours define in which direction the normal should be shifted, the 3 colour channels being mapped to the 3 directions X, Y and Z. This allows more detail and control over the effect.

185.15. O

- **Orange**²³ is the first Blender open movie project.
- **Oversampling** (OSA), also called **Anti-Aliasing** is the technique of minimizing aliasing when representing a high-resolution signal at a lower resolution. In most cases, anti-aliasing means removing data at too high a frequency to represent. When such data is left in a signal, it causes unpredictable artifacts.

²¹ <http://en.wikipedia.org/wiki/Del>

²² <http://en.wikipedia.org/wiki/Nonuniform%20rational%20B-spline>

²³ <http://orange.blender.org>

185.16. P

- **Phong** shading term is used indiscriminately to describe both an illumination model and an interpolation method in 3D computer graphics. Phong reflection is a local illumination model and can produce a certain degree of realism in three-dimensional objects by combining three elements - diffuse, specular and ambient for each considered point on a surface. It has several assumptions - all lights are points, only surface geometry is considered, only local modelling of diffuse and specular, specular colour is the same as light colour, ambient is a global constant.
- **Point Light** is a light that has a specific location and radiates equally out in all directions. Examples of point lights would be candles or bare lightbulbs. Surfaces close to the point light are brighter than those which are far away. Point lights have attenuation, which controls how quickly the light intensity drops off as you move away from it. Lights with high attenuation are very localized, while lights with low attenuation will spread farther.
- **Polygonization** (*of meta-surfaces*) is the process of approximating the meta-surface via polygons so it can be displayed/rendered in Blender.
- **Purple** runs as a normal Verse client. It implements a node database to mirror the contents of its host. It loads the plug-ins, which reside in libraries, from local disk as DLLs or shared objects depending on the platform.

185.17. Q

- **Quaternion**²⁴ is a representation of 3D rotations with four numbers. It can be interpreted as an extension of complex numbers to 3D. The interpretation of the four numbers is not very intuitive for a human, but the numerical advantage of quaternions is that it is the smallest mathematical representation that does not suffer from the gimbal lock²⁵ singularity²⁶ problem. This problem occurs for example with Euler angle²⁷ representations, when a small change in the 3D orientation can give rise to a large change in Euler angles.

185.18. R

- **Radiosity** is a more accurate but also more process-intensive technique than raytracing, that calculates patterns of light and shadow for rendering graphics images from three-dimensional models. One of the many different tools which can simulate diffuse lighting in Blender.
- **Raytracing** works by tracing the path taken by a ray of light through the scene, and calculating reflection, refraction, or absorption of the ray whenever it intersects an object in the world. More accurate than Scanline, but much slower.

²⁴ <http://en.wikipedia.org/wiki/quaternion>

²⁵ <http://en.wikipedia.org/wiki/gimbal%20lock>

²⁶ <http://en.wikipedia.org/wiki/mathematical%20singularity>

²⁷ <http://en.wikipedia.org/wiki/euler%20angles>

- **Render:** to generate actual viewable images from a 3D model or scene. This may or may not need to happen in real time; for example an interactive game requires real-time rendering, whereas a feature film does not. These situations require very different rendering techniques.
- **Refraction** in geometric optics is the change in direction of a wave due to a change in velocity. It happens when waves travel from a medium with a given refractive index to a medium with another. At the boundary between the media the wave changes direction; its wavelength increases or decreases but frequency remains constant. For example, a light ray will refract as it enters and leaves glass.
- **Relative Vertex Keys (RVK)** are part of a keyframe animation system that operates on vertex level objects. Each (shape) key is stored as a morph target such that several keys may be blended together to achieve complex mesh animation. With RVK you can create facial expressions, speech, and other detailed animated keyframed movements within your mesh-based models.
- **Rig:** the controls that aid in the manipulation of a digital character.
- **Rigging** is the process by which a person creates constraints and relationships between objects that will generate controls to aid in the manipulation of a digital character.

185.19. S

- **Scanline** is one row of pixels in the final render. Also the term for one of the methods of rendering which Blender can use. It is much faster than Raytracing, but allows fewer effects, such as reflections, refractions, motion blur and focal blur.
- **Shader:** an algorithm for computing the appearance of a given material based on the colour, angle and intensity of the light. *Specular* shaders produce a more shiny, mirrorlike finish, while *diffuse* shaders give a duller surface appearance. There are also *toon* shaders, which are deliberately designed to produce an effect more akin to a cartoon drawing, with delineated object borders and less gradation of colours over a surface.
- **Shadows:** simulated lights don't normally cast shadows. And, they also pass through solid objects - so a light inside a closed box would actually illuminate things outside the box as if the box were transparent. The shading on objects is only calculated based on the angle of the surface.
- **Specular Light** refers to the highlights on reflective objects, like diamonds, billiard balls, and eyes. Specular highlights often appear as bright spots on a surface, at a point where the light source hits it directly. Ambient, Diffuse, and Specular are called the three components of a light source. Each one is given a color, which, when added together, create the final color of a light. For most lights, the main overall color of the light is defined by the Diffuse color. Sunlight or lightbulbs would be white, while moonlight would be a darker blue, and a candle would be yellow. You can use the ambient color to adjust the overall color range of the light source; or, you can get a slight tint to shadows by making the diffuse component yellow and the ambient a slight blue. In many lights, the ambient color is left at black, meaning that it won't have any effect. Specular components are often left at white, but you can make them different colors to get interesting effects. Most of the time you can completely ignore the specular and diffuse settings on a light, but just be aware that the way you set the color is by specifically setting the diffuse color. The final color that an object appears to be is a combination of the light hitting it and the color of the surface.

- **Spotlight** is a light with both location and direction. A spotlight sends out a cone of light defined by the spotlight angle, and illuminates only objects within that cone. Spotlights also have attenuation, as well as a parameter that controls whether the spot of light is sharply defined or has smooth edges. These 4 types of lights are listed in order of computational complexity; the more lights you have, the more work the computer has to do. Generally it's a good idea to use directional lights whenever possible, since they're the cheapest, and use pointlights and spotlights sparingly.
- **Stucci** is one of the classes of blender textures. *Stucci* is not an English word, but is used in Blender as the plural of stucco²⁸.
- **Subdivision Surface** (Subsurf) is the tool which subdivides your model at render-time, without affecting your mesh at design-time. There are two subsurf algorithms in Blender to choose from - *Simple Subdiv*, which doesn't affect the shape of your mesh, and is used to add detail to displacement mapping or render-time radiosity, both of which operate on a per-vertex basis. The other is *Catmull-Clark*, a common subdivision algorithm which smooths out curves, and allows you to make complicated smooth surfaces (e.g. people, plants, etc.) with very few faces. However this algorithm can sometimes (read: often) have strange results with meshes containing triangles or vertices with many edges ("poles"), unless it is correctly handled.
- **Sub Surface Scattering** (SSS) is a mechanism of light transport in which light penetrates the surface of a translucent object, is scattered by interacting with the material, and exits the surface at a different point. All non-metallic materials are translucent to some degree. In particular, materials such as marble, skin, and milk are extremely difficult to simulate realistically without taking subsurface scattering into account.

185.20. T

- **Tuhopuu** is an experimental version of Blender that is like a code playground, developers can put their new code in there to be tested and played with by users before it gets put into the official Blender. Tuhopuu is Finnish for "Tree of destruction".
- **Tweening** is short for in-betweening, the process of generating intermediate frames between two images to give the appearance that the first image evolves smoothly into the second image. Tweening is a key process in all types of animation, including computer animation. Sophisticated animation software enables one to identify specific objects in an image and define how they should move and change during the tweening process. Another word for tweening is interpolation²⁹.

185.21. U

- **UV Mapping** (UV) This refers to the process of re-parameterizing a 3d object with dimensions x, y and z into a 2d plane with coordinates u and v. Most texturing requires this step because it tells the program HOW to apply a 2d image map onto a 3d object. If all your textures must be 2d and flat, the easiest way to determine what pixel goes where is if your model is flattened and made 2d. It also establishes a relationship between a 2d image and the mesh such that if the mesh deforms,

²⁸ <http://en.wikipedia.org/wiki/stucco>

²⁹ <http://en.wikipedia.org/wiki/interpolation>

the image map will deform along with it. Think of it as skinning a cat and pinning its hide onto cardboard to facilitate painting it!

185.22. V

- **Verse** is a network protocol that lets multiple applications act together as one large application by sharing data over a network. If one application makes a change to shared data, the change is distributed instantly to all the other interested clients.

185.23. W

- **WC** is Weekend Challenge.
- **WIP** is Work In Progress.

185.24. X

185.25. Y

- **Yet Another Free Raytracer** (YafRay) is an open source ray tracing program that uses an XML scene description language. It has been integrated into, and is often used to render scenes made in, Blender.

185.26. Z

186. FAQ

The following questions cover general information about *Blender 3D: Noob to Pro* and the software it is designed to teach, *Blender*.

186.0.1. What is Blender?

Blender is a full-blown 3D content creation suite, usable for 3D editing, animation, texture mapping, compositing and rendering.

It runs on:

- Windows — XP, Vista, and 7, 32- and 64-bit
- Mac OS X — PowerPC and Intel, 32- and 64-bit
- Linux — Intel x86 and AMD, 32- and 64-bit
- BSD
- Solaris
- Irix.

It is free, open-source software, available from blender.org¹.

Blender **2.58a** is the latest stable release as of 16/07/11. Beginners are urged to learn using stable 2.5x versions, as there has been a significant UI overhaul since 2.49 and future tutorials will likely be written or recorded with the new interface. However, keep in mind that many existing tutorials were written for 2.49. If you're having trouble finding a button or tool that has moved, try pressing Spacebar to search for Blender functions.

For more about Blender's history, functionality, and media productions which have utilized Blender, see the Blender entry in Wikipedia².

186.0.2. What is this book? And what isn't it?

Noob to Pro is a collection of tutorials. Following these tutorials you can go from ignorant new Blender user (noob) to proficient Blender power user (pro). Though the tutorials are not strictly linked, we recommend going through them in order. Earlier tutorials introduce skills and information you'll be expected to know in later sections.

Noob to Pro isn't a compendium or reference manual. Use it with other resources, including those in Blender's Help menu. *Noob to Pro* provides links to outside resources. We're part of the Blender community!

1 <http://www.blender.org>

2 <http://en.wikipedia.org/wiki/Blender%20%28software%29>

186.0.3. How accurate is the information in this book?

Blender is still under development! Its appearance, tools, and capabilities change from time to time. For instance, Blender 2.5x alters both the graphical user interface and the Python scripting interface and may also change some keyboard shortcuts.

Most of the tutorials should work with any recent version of Blender. *Blender 3D: Noob to Pro* readers are encouraged to edit the WikiBook in order help to keep it up to date.

186.0.4. I have a question, where can I get help?

Search or post questions at BlenderArtists³ and Blender Newbies forum⁴, both large, easy-to-use, English-language forums. BlenderWiki⁵ contains a large database of documentation. Find similar forums in other languages at blender.org's community page⁶. The discussion pages there are also useful.

If your question turns out to be very common, please take a moment to see if it can be added to the book! It is best if it can be smoothly incorporated with an existing page.

3 <http://www.blenderartists.org/forum>

4 <http://forums.blendernewbies.com/>

5 <http://wiki.blender.org/>

6 <http://www.blender.org/community/user-community/>

187. Tutorial Links List

Here you can find useful **Blender¹ Tutorial Links** written in English. Feel free to add some cool tutorials, but, on this page, only in English. For tutorials in other languages please read [About²](#). A tutorial about editing these pages can be found [here³](#). New to the 3D world? Read [To Those Learning 3D⁴](#).

187.1. Official Blender Documentation

Almost all **Manuals** can be found in the latest official **Blender Manual Books**. For additional tutorials look below this topic.

- Tons of Video Tutorials⁵
- Blender MediaWiki⁶ * exposed
 - Documentation at Blender3D.org⁷
 - Categorized list of Blender 3D Tutorials⁸
 - official Blender wiki⁹

187.2. Interface

- The best website to learn Blender 3d¹⁰
- Blender Interface Theme Repository¹¹
- Basic Editing¹²
- Blender Hotkeys¹³
- Blender Hotkeys II¹⁴
- Basic Blender¹⁵ (Broken Link)

1 <http://en.wikipedia.org/wiki/Blender%20%28software%29>
2 Chapter 187.22 on page 1360
3 <http://en.wikipedia.org/wiki/Wikipedia%3AIntroduction>
4 http://glenmoyes.com/articles/to_those_learning_3d.shtml
5 <http://nystic.com/blender.php>
6 http://mediawiki.blender.org/index.php/Main_Page
7 <http://www.blender.org/download/documentation/>
8 http://www.surfnlearn.com/view_tutorials/Blender_3D/All/
9 <http://wiki.blender.org/>
10 <http://www.animationforcreation.com/>
11 <http://www.graphicall.org/themes/>
12 http://www.ee.oulu.fi/~kapu/cg_course/bigfiles/blender/blendman.html
13 <http://en.wikibooks.org/wiki/Blender%203D%3A%20HotKeys>
14 <http://blender.excellentwhale.com/>
15 http://www.planetannihilation.com/terrigen/tut_basic.shtml

- Basic Interface¹⁶ (Broken Link)
- Blender Multimedia¹⁷
- Changing Views¹⁸
- First Impression¹⁹
- Getting Started²⁰
- Removing Orphan Vertices²¹
- Seeing Textures with Alt-Z²²
- Stereoscopic camera for editing²³
- Blueprint Setup : Blender3d Club²⁴

187.3. Mesh Modeling

- Basic Modeling²⁵
- Video - DupliVerts Tutorial ²⁶
- How to solve Blender's smoothing problems²⁷: discussion about smoothing strategies: Smooth/-Solid, Auto Smooth, EdgeSplit modifier...
- Heightmaps²⁸
- Mesh Editing Techniques²⁹ *broken link*
- Object Extrusion and Procedural Objects³⁰ *broken link*
- Loops Corkscrews Problem³¹
- Loopings Problem³²
- Knots in Curves Problem³³
- ZeTool³⁴
- DupliFrames Modeling I³⁵

-
- 16 http://www.users.bigpond.net.au/blendage/pages/beginners_tutorials/tut1/basic_interface.html
- 17 <http://www.malefico3d.org/viejos-tutoriales/audiovideo-en.html>
- 18 <http://blender3dfr.free.fr/anglais/tut2/tut2.htm>
- 19 <http://blender3dfr.free.fr/anglais/premiere.htm>
- 20 <http://blender3dfr.free.fr/anglais/tut1/tut1.htm>
- 21 http://membres.lycos.fr/bobois/Tuts/reperer_sommets_orphelins/sommets_orphilins.html
- 22 http://membres.lycos.fr/bobois/Tuts/subdiviser_pour_conquerir_la_texture/subdiviser_pour_conquerir_la_texture.html
- 23 <http://brunetton.tuxfamily.org/index.php?n=En.BlenderStereo>
- 24 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=6>
- 25 <http://www.animationforcreation.com/2008/09/lesson-two-basic-of-blender.html>
- 26 <http://www.youtube.com/watch?v=M6EPXc8N3qs>
- 27 <http://feblemind.tuxfamily.org/blog/index.php/post/2007/08/13/88-how-to-solve-blenders-smoothing-problems>
- 28 http://members.tripod.com/~funky_munky/tuts/blender/heightmaps.htm
- 29 <http://vrotvrot.com/xoom/tutorials/Corridor/Corridor.html>
- 30 <http://vrotvrot.com/xoom/tutorials/mineRide/mineride.html>
- 31 http://membres.lycos.fr/bobois/Tuts/Courbes/tire_bouchon/tire-bouchon.html
- 32 <http://membres.lycos.fr/bobois/Tuts/Courbes/Looping/looping.html>
- 33 http://membres.lycos.fr/bobois/Tuts/Courbes/Aplat/a_plat.html
- 34 <http://membres.lycos.fr/bobois/Tuts/Courbes/ZeTool/zetool.html>
- 35 http://membres.lycos.fr/bobois/Tuts/DupliFrames/Modeling/modeling_with_duplifames_en_1.html

- DupliFrames Modeling II³⁶
- Dupliframes³⁷
- Spin Modeling : Blender3d Club³⁸
- Spin and Spin Dup³⁹ - Needs a lot more explanation
- Automatized BevOb⁴⁰
- Extrusion along a Path using BevOb⁴¹
- Path Extrusion⁴²
- DupliVerts⁴³
- Limit DupliVerts⁴⁴
- Types of Handles for Bezier Curves⁴⁵
- Beveled Cube⁴⁶ (*pdf tutorial*)(Broken Link)
- Joining/Separating Parts of a Mesh⁴⁷
- Extrusion Controlled by IPO⁴⁸
- Basic objects⁴⁹
- Modelling Techniques and Strategies⁵⁰
- Easily Remove Orphan Edges and Vertices⁵¹
- Non-destructive bevel effect⁵²
- "Cage" Technique Tutorial⁵³
- Creating Fire⁵⁴

-
- 36 http://membres.lycos.fr/bobois/Tuts/DupliFrames/Modeling/pizza_boxes/stack_of_boxes_en.html
- 37 http://membres.lycos.fr/bobois/Tuts/DupliFrames/les_bases/les_bases_en_1.html
- 38 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=45>
- 39 http://membres.lycos.fr/bobois/Tuts/Spin_et_spin_dup/Spin_and_spin_dup_tutorial.html
- 40 http://membres.lycos.fr/bobois/Tuts/Extrusion_par_bev_ob/Ext1_Ext2/Ext1_Ext2_en.html
- 41 http://membres.lycos.fr/bobois/Tuts/Extrusion_par_bev_ob/Alignement_des_axes/Alignement_des_axes_en.html
- 42 http://membres.lycos.fr/bobois/Tuts/Extrusion_par_bev_ob/Les_bases/Les_bases_en.html
- 43 http://membres.lycos.fr/bobois/Tuts/DupliVerts/Dupliverts/dupliverts_en.html
- 44 http://membres.lycos.fr/bobois/Tuts/DupliVerts/limiter_les_dupliverts_en.html
- 45 http://membres.lycos.fr/bobois/Tuts/Courbes/Courbes_de_bezier/Types_de_poignees_bezier/Types_de_poignees_bezier_en.html
- 46 <http://w1.185.telia.com/~u18510119/tutorials/makeacube.pdf>
- 47 http://membres.lycos.fr/bobois/Tuts/joindre_et_separer/joindre_et_separer_ang.html
- 48 http://jmsoler.free.fr/didacticiel/blender/tutor/en_modelextipo00.htm
- 49 <http://blender3dfr.free.fr/anglais/tut3/tut3.htm>
- 50 <http://www.elysiun.com/forum/viewtopic.php?t=21382>
- 51 http://jmsoler.free.fr/didacticiel/blender/tutor/modesel_somparaz_en.htm
- 52 <http://blendertips.blogspot.com/2006/03/bevel-modifier-workaround.html>
- 53 <http://www.cyphertxt.com/blendertechniquetut.php>
- 54 <http://faintmedia.com/forums/topic/95-fire-ball/>

187.4. Nurbs and Subsurface Modeling

- Subsurface modeling tutorial⁵⁵
- IPOs, Lattices, Nurbs & Stuff⁵⁶ (Broken Link)
- Cross-Sections⁵⁷
- Curve Resolution⁵⁸
- Weight Parameter⁵⁹
- OrderU, UniformU and EndpointU⁶⁰
- Curves -n- Bevels⁶¹
- Making a Hole in Subsurf⁶²
- Subsurf Edge Rolling - Round⁶³
- Subsurf Edge Rolling - Square⁶⁴
- Modeling with lattices - a fork⁶⁵
- Subsurf modeling 2⁶⁶
- Metaballs⁶⁷
- Abstract SubSurf modeling I⁶⁸

187.5. Specific Object Modeling

- Sponge Bob : Blender3d Club⁶⁹
- Double-Helix⁷⁰
- Turtle⁷¹
- Leaf Shader Tutorial⁷²

-
- 55 <http://www.animationforcreation.com/2008/10/lesson-6-imageseparating-objects.html>
- 56 http://dev.newmediaworx.com/johnnyb/bplanet/tut_bottle.htm
- 57 <http://www.blendernation.com/tutorials/blender-3d-beginner-tutorial-the-cave-of-torsan-a/>
- 58 http://membres.lycos.fr/bobois/Tuts/Extrusion_par_bev_ob/DefResolU/DefResolU_en.html
- 59 http://membres.lycos.fr/bobois/Tuts/Courbes/Courbes_nurbs/Poids_des_voisins/Poids_des_voisins_en.html
- 60 http://membres.lycos.fr/bobois/Tuts/Courbes/Courbes_nurbs/Influence_de_order/Influence_de_order_en.html
- 61 <http://www.ingiebee.com/Blendermania/curves%20and%20Bevels.htm>
- 62 http://membres.lycos.fr/bobois/Tuts/subsurfs/Trouer_une_surface/trouer_une_subsurf_ang.html
- 63 http://membres.lycos.fr/bobois/Tuts/subsurfs/rouler_un_bord/rouler_un_bord_rond_ang.html
- 64 http://membres.lycos.fr/bobois/Tuts/subsurfs/rouler_un_bord/rouler_un_bord_carre_ang.html
- 65 <http://www.blender3d.org/documentation/htmlI/x10279.html>
- 66 <http://blender3dfr.free.fr/anglais/tut5/tut5.htm>
- 67 <http://blender3dfr.free.fr/anglais/tut6/tut6.htm>
- 68 <http://www.deviantart.com/view/26691152/>
- 69 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=61>
- 70 <http://www.deviantart.com/view/27132715/>
- 71 <http://www.elysiun.com/forum/viewtopic.php?t=44686>
- 72 <http://www.elysiun.com/forum/viewtopic.php?t=39810>

- Sears-Roebuck Dairy Barn⁷³
- Golf Ball⁷⁴
- Celtic Knot⁷⁵
- Simple Tree⁷⁶
- Making a pedestal⁷⁷ (broken)
- Building a Castle⁷⁸ (broken)
- Ice Cube⁷⁹ (*pdf tutorial*)
- Water I⁸⁰ (*pdf tutorial*)
- Water II⁸¹ (*pdf tutorial*)
- Space Pod⁸² (Dead link)
- Grass⁸³ (Link moved)
- Bongo Creature⁸⁴ (Broken Link)
- Spiral Stairs⁸⁵ (All blenderwars links dead)
- Waste Basket⁸⁶ (Domain Expired)
- Beveled Cube⁸⁷ (Domain Expired)
- Simple Box⁸⁸ (Domain Expired)
- Fountain with Moving Water⁸⁹ (Dead link)
- Track Creation⁹⁰ (Broken link)
- Dolphin⁹¹ (dead link)
- Dice⁹²
- Die with round corners⁹³
- Logo⁹⁴
- Cutting through steel⁹⁵
- Roller Coaster⁹⁶
- Roller Coster Cobra⁹⁷

73 <http://www.harkyman.com/searsbarn01.html>

74 <http://www.elysiun.com/forum/viewtopic.php?t=31781&postdays=0&postorder=asc&start=15>

75 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_celticknot_en.htm

76 <http://www.elysiun.com/forum/viewtopic.php?t=32306>

77 http://web.telia.com/~u91121962/tut1_1.htm

78 http://www.blender3d.org/_media/education/quickstart/Building_Castle.html

79 <http://caron.yann.free.fr/blender/IceCube.pdf>

80 <http://www.selleri.org/Blender/tuts/Water.pdf>

81 <http://www.selleri.org/Blender/tuts/Water2.pdf>

82 http://dev.newmediaworx.com/johnnyb/bplanet/tut_pod.htm

83 <http://lib.hel.fi/~basse/>

84 <http://www.enricovalenza.com/makebongo.html>

85 <http://www.blenderwars.com/tut.php?module=stairs>

86 <http://www.blenderwars.com/tut.php?module=waste>

87 <http://www.blenderwars.com/tut.php?module=bevcube>

88 <http://www.blenderwars.com/tut-path.php?module=box1>

89 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut5/fountain.html>

90 <http://www.geocities.com/swdoughty/blendertut1.html>

91 <http://www.blendernation.com/tutorials/blender-3d-beginner-tutorial-dolphin/>

92 <http://www.blendernation.com/tutorials/blender-3d-beginner-tutorial-a-die/>

93 <http://www.youtube.com/watch?v=sp53YwtWdfY/>

94 <http://www.blendernation.com/tutorials/blender-3d-beginner-tutorial-creating-a-logo/>

95 <http://www.blendernation.com/tutorials/blender-3d-intermediate-tutorial-cutting-through-steel/>

96 http://membres.lycos.fr/bobois/Tuts/Courbes/completeRC/foreword_en.html

97 <http://membres.lycos.fr/bobois/Tuts/Courbes/Cobra/cobra.html>

- Modelling a Glass⁹⁸ (broken)
- Ocean View⁹⁹
- Solar Systems for beginners¹⁰⁰ (Domain Expired)
- Chair¹⁰¹
- Spiral Stair¹⁰² (Broken link)
- Lighthouse¹⁰³ (Dead link)
- Landscape¹⁰⁴ (Broken link)
- Landscapes Easy¹⁰⁵ (Dead link)
- Cloth Tutorial¹⁰⁶ (dead link) archived version¹⁰⁷
- Cup¹⁰⁸
- Cola cane¹⁰⁹ (Dead Link)
- Candle¹¹⁰
- Volcano¹¹¹
- Orbital Logo¹¹²
- Realistic Planet¹¹³ (Dead Link)
- Landscape Cartoon¹¹⁴ (*pdf tutorial*)
- Text¹¹⁵
- Text Part 1: Manipulate Text¹¹⁶ (Old link. Might this be the aimed for one? Text Part 1: Manipulate Text¹¹⁷)

187.5.1. Human

- Hand Palm¹¹⁸
- Eyes¹¹⁹

98 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/tp1-index-ang.html

99 <http://www.ingiebee.com/Blendermania/create%20ocean%20view.htm>

100 <http://www.ingiebee.com/Blendermania/Solar%20System.htm>

101 http://membres.lycos.fr/bobois/objets_a_partager/a_simple_chair.html

102 <http://w1.185.telia.com/~u18510119/tutorials/spiralstair.pdf>

103 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut1/lighthouse.html>

104 <http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/paysages/index-ang.html>

105 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut2/landscape.html>

106 <http://www.free-webspace.biz/sonix/BlenderTutes/QuickClothTute.html>

107 <http://web.archive.org/web/20040805214133/http://www.free-webspace.biz/sonix/BlenderTutes/QuickClothTute.html>

108 <http://kahuna.clayton.edu/~jbrooks/blender/tutorials/cup/>

109 <http://free.hostdepartment.com/B/Blender3D/cantut/>

110 http://download.blender.org/documentation/oldsite/oldsite.blender3d.org/93_Blender%20tutorial%20Candle.html

111 <http://kahuna.clayton.edu/~jbrooks/blender/tutorials/volcano/>

112 <http://kahuna.clayton.edu/~jbrooks/blender/tutorials/universal/>

113 <http://www.enricovalenza.com/realplan.html>

114 <http://www.selleri.org/Blender/tuts/CartoonishLandscape.pdf>

115 <http://blender3dfr.free.fr/anglais/tut4/tut4.htm>

116 <http://www.raivestudios.com/tutorials/blender/text/>

117 <http://www.faintmedia.com/forums/topic/93-text/>

118 <http://jlp.nerim.net/tutorials/hands-modelling/hands-tuto-01.html>

119 <http://web.pdx.edu/~wlf/tut.html>

- Hair Tutorial¹²⁰
- Wikipedia: Human Body Proportions¹²¹

187.5.2. Cars

- Creating a Toon Car¹²² (Creating a Toon Car¹²³ , The old, Broken Link)

187.6. Texture Mapping

- Video - Basic Materials¹²⁴
- Making a Lightsaber with Halos (link repaired!)¹²⁵
- The Unofficial Texturing Tutorial¹²⁶
- Material Indices Tutorial¹²⁷
- Transparency and Refraction¹²⁸
- Reflections¹²⁹
- Texturing landscapes according to slopes¹³⁰
- Using Texture Stencils¹³¹
- Chrome and Shiny Metal Surfaces¹³²
- Texturing Tutorial¹³³ (NOTE: For the game Total Annihilation, not really a general tut)
- Textures in OrCo mode¹³⁴(Broken link)
- Material Indices¹³⁵(Broken link)
- Textures Channels¹³⁶(Broken link)

120 <http://web.archive.org/web/20031230091345/http://www.elysiun.com/forum/viewtopic.php?t=16008>

121 http://en.wikipedia.org/wiki/Body_proportions

122 <http://www.faintmedia.com/forums/topic/84-toon-car/>

123 <http://www.raivestudios.com/tutorials/blender/tooncar/>

124 <http://www.youtube.com/watch?v=2kk2BkVQkyk>

125 <http://blenderartists.org/forum/showthread.php?t=62611>

126 <http://www.elysiun.com/forum/viewtopic.php?t=11889>

127 <http://feeblemind.tuxfamily.org/blog/index.php?post/2004/12/21/5-blender-indices-materiau-material-indice>

128 <http://feeblemind.tuxfamily.org/blog/index.php?post/2004/12/26/7-didacticiel-la-refraction-tutorial-refraction>

129 <http://feeblemind.tuxfamily.org/blog/index.php?post/2004/12/26/6-didacticiel-les-reflets-tutorial-reflections>

130 <http://feeblemind.tuxfamily.org/blog/index.php?post/2009/10/12/113-texturing-landscapes-according-to-slopes>

131 <http://feeblemind.tuxfamily.org/blog/index.php?post/2005/02/27/16-didacticiel-usage-des-masques-de-textures-tutorial-using-texture-stencils>

132 http://www.ualberta.ca/%7Enwy/blender/blender_tut1.htm

133 http://www.planetannihilation.com/terrigen/tut_buildpics.shtml

134 <http://www.enricovalenza.com/textures.html>

135 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/tp2-index-ang.html

136 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/didac4-ang.html

- Textures Mapping¹³⁷(Broken link)
- Specular Color¹³⁸(Broken link)
- The Color¹³⁹(Broken link)
- Materials in Blender¹⁴⁰(Broken link)
- Alpha, Bump, and Specular image textures¹⁴¹
- Specular Maps : Blender3d Club¹⁴²
- Image Textures¹⁴³ (Broken link)
- Decals¹⁴⁴ (Broken link)
- Chrome¹⁴⁵ (Broken link)
- Alpha Masks¹⁴⁶
- Textures with Alpha¹⁴⁷
- Textures with Bumpmapping¹⁴⁸
- Using More Than One Color¹⁴⁹(Broken link)
- Adding Color To Your Shapes¹⁵⁰(Broken link)
- Using Textures and Bumpmapping Them¹⁵¹(Broken link)
- Texture Mapping Tutorial¹⁵² (*pictures ok now 18/03/2006*)(Broken link)
- Environment Mapping¹⁵³ (*pictures ok now 18/03/2006*)(Broken link)
- Mapping Character¹⁵⁴(Broken link)
- Creation and Mapping¹⁵⁵
- Displacement Mapping¹⁵⁶(Broken link)

137 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/didac3-ang.html

138 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/didac2-ang.html

139 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/didac1-ang.html

140 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/blender_material/index-ang.html

141 <http://www.ingiebee.com/Blendermania/BumpSpecAlphaTex.htm>

142 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=51>

143 <http://www.ingiebee.com/Blendermania/attach%20image%20file%20to%20object.htm>

144 <http://www.ingiebee.com/tutorials/Decal%20Mirror%20Modelling/theeth%20decal.htm>

145 <http://www.ingiebee.com/tutorials/Digital-Mark.htm>

146 http://membres.lycos.fr/bobois/Tuts/trouer_par_texture/trouer_par_texture.html

147 <http://www-users.cs.umn.edu/~mein/blender/tutorials/dust/alpha.html>

148 <http://www-users.cs.umn.edu/~mein/blender/tutorials/dust/bump.html>

149 http://www.users.bigpond.net.au/blendage/pages/tutorials/tut9/two_colours.html

150 http://www.users.bigpond.net.au/blendage/pages/beginners_tutorials/tut3/adding_colour.html

151 http://www.users.bigpond.net.au/blendage/pages/beginners_tutorials/tut4/adding_texture.html

152 http://dev.newmediaworx.com/johnnyb/bplanet/tut_mapping.htm

153 http://dev.newmediaworx.com/johnnyb/bplanet/tut_envmap.htm

154 <http://otothegardener.free.fr/tutorials/LittleOTO/anonce/anonce.htm>

155 http://membres.lycos.fr/bobois/Tuts/boite_de_projection/boite_de_projection005_en.html

156 <http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/displacement/index-ang.html>

- Game Sprites¹⁵⁷ (Broken link)
- Stencilling Textures¹⁵⁸ (Broken link)

187.6.1. UV Mapping

- Video - UV Mapping¹⁵⁹
- UV Mapping & Texturing¹⁶⁰
- Sub-Surf and UV-Texturing problem¹⁶¹
- UV Mapping¹⁶²
- Complex material for a sword¹⁶³

187.6.2. Animated Textures

- Video - Changing Colors Of Objects Mid Animation¹⁶⁴
- Blender TexMesh Tutorial¹⁶⁵ * exposed
 - animating masks for simulating ice freeze¹⁶⁶
 - Animated Procedural Textures¹⁶⁷ (*pdf tutorial*)

187.6.3. Nodes

- Video - Blender Nodes¹⁶⁸
- Vector Blur usage with Blender¹⁶⁹

187.7. 2D Texture Painting Techniques

This part isn't about using Blender, but about necessary **2D knowledge** for advanced Blender users.

- Steven Stahlberg's Tutorials¹⁷⁰ * exposed
 - Dirty Metal Texture¹⁷¹ (Broken Link)

157 <http://www.ingiebee.com/Blendermania/Game%20Sprites.htm>

158 <http://www.ingiebee.com/Blendermania/Stencilling.htm>

159 <http://www.youtube.com/watch?v=mVhhZ2KRr4Q>

160 http://biorust.com/index.php?page=tutorial_detail&tutid=85

161 http://barney.gonzaga.edu/~amoore1/uv_mapping_project/

162 <http://membres.lycos.fr/bobois/Tuts/uvmapping/uvmapping01.html>

163 http://www.3dtotal.com/team/Tutorials/swords&daggers/swords&daggers_01.asp

164 <http://www.youtube.com/watch?v=Eljv69HOLR4>

165 <http://www.telusplanet.net/public/kugyelka/blender/tutorials/texmesh/texmesh.htm>

166 <http://www.elysiun.com/forum/viewtopic.php?p=455683#455683>

167 <http://www.cogfilms.com/tutorials.html>

168 <http://www.youtube.com/watch?v=-tM1spWQ5aw>

169 <http://Feeblemind.tuxfamily.org/blog/index.php?post/2006/11/05/74-blender-and-the-vector-blur>

170 <http://www.androidblues.com/howto.html>

171 <http://div.dyndns.org/EK/tutorial/texture/>

- Gritty Pipelines¹⁷² (Broken Link)
- Digital Painting¹⁷³ (Broken Link)
- NWN Texturing Tutorials¹⁷⁴
- Fixing Lighting Irregularities in Self-Tiling Maps¹⁷⁵
- Texture Map for the Iris of the Eye¹⁷⁶ (Broken Link)
- Airplane Texture Tutorial¹⁷⁷ (Broken Link)
- Game Skinning Tips¹⁷⁸ (Broken Link)
- Game Lighting Basics I¹⁷⁹ (Broken Link)
- Game Lighting Basics II¹⁸⁰ (Broken Link)
- Texturing with Gimp¹⁸¹

187.8. Lighting, Shadows and Rendering

- Light - a detailed tutorial¹⁸² (Broken Link)
- Yafray as an Integrated External Renderer¹⁸³ (Broken Link)
- Basic 3 Point Lighting¹⁸⁴
- Radiosity I¹⁸⁵
- Radiosity II¹⁸⁶
- Ramp Shaders¹⁸⁷
- Simulating Radiosity¹⁸⁸
- Caustic Sampler Tutorial¹⁸⁹ (Broken Link)
- Shadows Control¹⁹⁰ (Broken Link)
- The 'World' buttons within Blender¹⁹¹ (Broken Link)
- Mastering Shadows¹⁹² (Broken Link)

172 <http://div.dyndns.org/EK/tutorial/gritty/>

173 http://www.planetquake.com/polycount/resources/general/tutorials/HitmanDaz_Tut01/DigitalPainting.shtml

174 http://www.btinternet.com/~i.nation/tutorials/tutorial_00.htm

175 <http://www.3drender.com/light/EqTutorial/tiling.htm>

176 <http://www.kandsdesign.com/kim/eyemap-tut.html>

177 <http://www.geocities.com/SiliconValley/Haven/2470/txttrtut/txttrtut.html>

178 <http://www.planetquake.com/pandemonium/html/skintips.htm>

179 <http://www.planetquake.com/pandemonium/html/skintut3.htm>

180 <http://www.planetquake.com/pandemonium/html/skintut4.htm>

181 <http://otothecleaner.free.fr/tutorials/GimpTexture/gimptexture.htm>

182 <http://www.itchy-animation.co.uk/tutorials/light01.htm>

183 <http://wiki.yafray.org/bin/view.pl/UserDoc/GauravGuide>

184 <http://www.andrew-whitehurst.net/3point.html>

185 <http://blenderman.free.fr/tut/radiosity/uk/>

186 http://download.blender.org/documentation/html/chapter_rendering_radiosity.html

187 <http://www.blender.org/development/release-logs/blender-234/ramp-shaders/>

188 http://jmsoler.free.fr/didacticiel/blender/tutor/lumiere_radios_en.htm

189 <http://www.elysiun.com/forum/viewtopic.php?t=30265>

190 http://dev.newmediaworx.com/johnnyb/bplanet/tut_shadows.htm

191 <http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/world/index-ang.html>

192 <http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/shadows/index-ang.html>

- Ambient Occlusion¹⁹³
- Soft Shadows and Area Lights¹⁹⁴
- Depth of Field¹⁹⁵ (Broken Link)
- Soft Lights¹⁹⁶ (*pdf tutorial*)
- Blender's Mist¹⁹⁷
- Basic Lighting¹⁹⁸
- Optimizing Renders¹⁹⁹
- Flares²⁰⁰
- Toon Shading²⁰¹ (Broken Link)
- Simple Gold Ring with Caustics²⁰² (Broken Link)

187.9. Armatures and IK

- Video - Basic Armatures²⁰³
- Dancing Flor²⁰⁴ * exposed
 - Animation Workshop II²⁰⁵ * exposed
 - Using Ipo driven shape keys to correct deformations in joints²⁰⁶ * exposed (Broken Link)
 - Animation recode project²⁰⁷
 - Action Constraints tutorial made easy²⁰⁸
 - Rigged Character²⁰⁹ (broken link)
 - Driven Hand Rig²¹⁰
 - Constrained Mechanics²¹¹
 - Armatures Tutorial²¹² (Broken Link)
 - Animation using Armatures²¹³ (Broken Link)

193 <http://feeblemind.tuxfamily.org/blog/index.php?post/2005/01/16/10-didacticiel-locclusion-ambiante-tutorial-ambient-occlusion>

194 <http://feeblemind.tuxfamily.org/blog/index.php?post/2005/01/02/9-didacticiel-les-aires-lumineuses-tutorial-area-lights>

195 <http://www.malefico3d.com.ar/tutor/dof-en.html>

196 <http://www.selleri.org/Blender/tuts/SoftLight.pdf>

197 <http://www-users.cs.umn.edu/~mein/blender/tutorials/dust/mist.html>

198 http://www.katorlegaz.com/tutorials/basic_3d_lighting.php

199 <http://www.katorlegaz.com/index.php?a=article&display=1078185600.php>

200 <http://www.centraresource.com/blender/flares/>

201 <http://free.hostdepartment.com/a/aner/TUTORIAL.html>

202 http://www.geocities.com/sound_man_dave/caustics.htm

203 <http://www.youtube.com/watch?v=GbFvriLssNo>

204 <http://otothecleaner.free.fr/tutorials/Flor/flor.html>

205 <http://www.elysiun.com/forum/viewtopic.php?p=511798>

206 <http://kokcito.tk/rvk/ipo.html>

207 <http://www.blender.org/development/release-logs/blender-240/how-armatures-work/>

208 <http://www.elysiun.com/forum/viewtopic.php?t=49603>

209 <http://www.elysiun.com/forum/viewtopic.php?t=31839>

210 <http://www.elysiun.com/forum/viewtopic.php?t=19347>

211 <http://argoslabs.com/%7Emalefico/tutor/mecano-en.html>

212 <http://www.anycities.com/turbog/Tutorial11.html>

213 http://www.users.bigpond.net.au/blendage/pages/tutorials/tut4/animatin_armatures.html

- [Ikas Blender 3D - Introduction](#)²¹⁴
- [Constrained Mechanics](#)²¹⁵ (Broken Link)
- [Using Armatures](#)²¹⁶ (Broken Link)
- [Making and Using Armatures](#)²¹⁷ (Broken Link)
- [Character Animation tools](#)²¹⁸ (Broken Link)
- [IK Solver Constraint](#)²¹⁹ (Broken Link)
- [Softbody for Rigged Characters](#)²²⁰ (Broken Link)
- [Rigging limbs that can twist](#)²²¹ (Broken Link)
- [Rigging tricks](#)²²² (Broken Link)
- [IKA Tutorial](#)²²³

187.10. Animation

- [Weight Painting : Blender3d Club](#)²²⁴
- [Learn Animation](#)²²⁵
- [Video - Animation Basics](#)²²⁶
- [Simple Animations](#)²²⁷
- [Basic Keyframing](#)²²⁸ (Broken Link)
- [Effects](#)²²⁹
- [Character Animation](#)²³⁰ (Broken Link)
- [Run Cycle](#)²³¹ (Non Blender specific)
- [Lip Sync I](#)²³² Blender 2.23
- [Lip Sync II](#)²³³ (Broken Link)
- [Walking Blues](#)²³⁴
- [Non Linear Action Editor - NLA I](#)²³⁵

214 <http://mmaigrot.free.fr/didac-blender/ikas/ikas-eng/ik-intro.php>

215 <http://www.malefico3d.com.ar/tutor/mecano-en.html>

216 <http://www.elysiun.com/tutorials/animation/>

217 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut3/armatures.html>

218 <http://www.elysiun.com/tutorials/animation/>

219 <http://www.elysiun.com/forum/viewtopic.php?t=42310>

220 <http://www.enricovalenza.com/softb.html>

221 <http://kokcito.tk/tut/rig3.html>

222 <http://kokcito.tk/tut/rig1.html>

223 <http://kahuna.clayton.edu/~jbrooks/blender/tutorials/IKA/>

224 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=21>

225 <http://www.animationforcreation.com/2008/10/lesson-10blendrs-animation-tools.html>

226 <http://www.youtube.com/watch?v=FAZFFLmbD2k>

227 <http://feeblemind.tuxfamily.org/blog/index.php?post/2005/03/08/>

17-didacticiel-animations-simples-1ere-partie-tutorial-simple-animations-1st-part

228 http://www.users.bigpond.net.au/blendage/pages/beginners_tutorials/tut2/

Keyframeanimation.html

229 <http://download.blender.org/documentation/html/c1585.html>

230 http://dev.newmediaworx.com/johnnyb/bplanet/tut_charanim.htm

231 http://rodri.aniguild.com/tuto_run/run_en.php

232 <http://www.meloware.com/blender/lipsync.htm>

233 http://dev.newmediaworx.com/johnnyb/bplanet/tut_charanim2.htm

234 <http://argoslabs.com/~malefico/tutor/walking.html>

235 <http://mmaigrot.free.fr/didac-blender/nla/eng/index.php>

- Non Linear Action Editor - NLA II²³⁶
- Wave Effect²³⁷ (Broken Link)
- Build Effect²³⁸ (Broken Link)
- Animation effects²³⁹ (Broken Link)
- Plane Following a Path²⁴⁰ (Broken Link)
- Object Following a Path²⁴¹
- Cyclic Animation²⁴²
- Animating Materials²⁴³ (Broken Link)
- Walking Tutorial²⁴⁴
- Change Cameras during an Animation²⁴⁵
- Camera Tracking With Additional Roll Constraint²⁴⁶
- Shape Keys / PDF: Blender3d Club²⁴⁷
- Eye Tracking : Blender3d Club²⁴⁸
- Simple Armature Animation : Blender3d Club²⁴⁹

187.11. Particles

- Everything about particles²⁵⁰
- Video - Particle Basics²⁵¹
- Video - Grass with Particles²⁵²
- Video - Explosions with Particles and the Explode Modifier²⁵³
- Particle Interaction²⁵⁴
- Static Particles²⁵⁵

-
- 236 <http://argoslabs.com/~malefico/tutor/nla-en.html>
- 237 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/animation_effects/didac2-ang.html
- 238 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/animation_effects/didac1-ang.html
- 239 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/animation_effects/index-ang.html
- 240 <http://www.ingiebee.com/Blendermania/plane%20follow%20path.htm>
- 241 <http://www.ingiebee.com/tutorials/Path/theeth%20paths.htm>
- 242 http://membres.lycos.fr/bobois/Tuts/animation_cyclique/animation_cyclique_01_ang.html
- 243 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut7/colourchanging.html>
- 244 <http://www.fortunecity.com/skyscraper/true/947/id13.htm>
- 245 <http://www.elysiun.com/forum/viewtopic.php?t=42840&highlight=change+camera>
- 246 <http://www.elysiun.com/forum/viewtopic.php?t=46181>
- 247 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=33>
- 248 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=32>
- 249 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=24>
- 250 <http://www.animationforcreation.com/2008/10/lesson-11blender-particle-system.html>
- 251 http://www.youtube.com/watch?v=4y1N9yjCP_4
- 252 <http://youtube.com/watch?v=M9aG0xB19IM>
- 253 <http://www.youtube.com/watch?v=At3hIc-423c>
- 254 <http://www.blender.org/development/release-logs/blender-234/particle-interaction/>
- 255 <http://www.elysiun.com/forum/viewtopic.php?t=41356>

- Static Particle Effect²⁵⁶ (Broken Link)
- Particles Effect²⁵⁷ (Broken Link)
- Making a Fireplace²⁵⁸ (Broken Link)
- Making a Fountain With Particles²⁵⁹ (Broken Link)* Dust Particles²⁶⁰
- Flames²⁶¹
- Liquid²⁶²
- Explosions²⁶³
- Smoke²⁶⁴
- Making a Rain Effect²⁶⁵
- Setting a prey-predator relationship using Boids particles²⁶⁶
- Particle and field basics²⁶⁷

187.12. Fluid Simulation

- Video - Fluid Simulation²⁶⁸
- Fluid Simulation Basics²⁶⁹
- Basic Blender Fluid Simulation guide²⁷⁰

187.13. Compositing

- Matching Real Lighting²⁷¹
- CGI/Live Action Interaction²⁷² (Broken Link)
- Compositing CG and Live Action in Blender²⁷³

256 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/animation_effects/didac4-ang.html

257 http://www.linuxgraphic.org/section3d/blender/pages/didacticiels/animation_effects/didac3-ang.html

258 <http://www.geocities.com/blengine/fireplace1.html>

259 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut5/fountain.html>

260 <http://www-users.cs.umn.edu/~mein/blender/tutorials/dust/particle.html>

261 <http://www.centuralsource.com/blender/flames/>

262 <http://www.centuralsource.com/blender/liquid/>

263 <http://www.centuralsource.com/blender/explosions/>

264 <http://www.centuralsource.com/blender/smoke/>

265 <http://feeblemind.tuxfamily.org/blog/index.php?post/2004/12/24/4-blender-faire-pleuvoir-making-rain>

266 <http://feeblemind.tuxfamily.org/blog/index.php?post/2007/12/28/106-setting-a-prey-predator-relationship-using-boids-particles>

267 <http://www.deviantart.com/view/27104504/>

268 <http://au.youtube.com/watch?v=yMBx2pYiqnY>

269 <http://www.deviantart.com/view/27479896/>

270 <http://www.xedium3d.com/tutorials/fluid/>

271 http://www.andrew-whitehurst.net/fx_light.html

272 http://st102.startlogic.com/~blenderw/cms/index.php?option=com_content&task=view&id=212&Itemid=42

273 <http://www.weirdhat.com/blender/compositing2/>

187.14. Game Engine

- Rigid Body Constraints : Blender3d Club²⁷⁴
- Video - Blender Game Engine Basics²⁷⁵
- Video - Making Dominoes²⁷⁶
- Game Engine Development²⁷⁷
- New fully integrated game engine? ²⁷⁸
- Game Blender Documentation²⁷⁹ (Last modified 13. 07. 2001)
- [irc://irc.freenode.net/gameblender #GameBlender]
- Walkthrough Tutorial²⁸⁰
- Overlay Scenes²⁸¹
- Walking Tutorial²⁸²
- Make a Menu²⁸³
- Flipper²⁸⁴ (Broken Link)
- Armatures in the Game Engine²⁸⁵ (Broken Link)
- Multilevel Maze²⁸⁶ (Broken Link)
- Creating Boulders that make the map restart when hit²⁸⁷ (Broken Link)
- Mouse Look 1st Person. Options include capping, invert mouse pitch and adjust mouse sensitivity.²⁸⁸
- Viewports. Creating a split screen.²⁸⁹
- Python game functions. Sample Code Included. Updated to 2.48²⁹⁰
- Car Setup. Use Bullet Physics vehicle wrapper. ²⁹¹
- Using Blender to host/contain a storyboard²⁹²
- Blender Games, .blends, models, textures, tutorials, and more.²⁹³ (Broken Link)
- Python scripting for the Game Engine²⁹⁴ at cgmasters.net.

274 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=19>

275 http://www.youtube.com/watch?v=WGycLDs_dLg

276 http://youtube.com/watch?v=3kb_utKDu3o

277 <http://wiki.blender.org/index.php/GameEngineDev>

278 <http://www.blender.org/forum/viewtopic.php?t=6409&sid=868ed3b00fd68157b704e87f0cc1684a>

279 <http://web.archive.org/web/20011207173733/http://www.blender.nl/gameBlenderDoc/book1.html>

280 <http://www.blender.org/education-help/tutorials/tutorial-folder/3d-walkthrough/>

281 <http://www.fortunecity.com/skyscraper/true/947/id16.htm>

282 <http://www.fortunecity.com/skyscraper/true/947/id13.htm>

283 <http://www.fortunecity.com/skyscraper/true/947/id17.htm>

284 <http://otothegardener.free.fr/tutorials/Flipper/flipper.htm>

285 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut8/realtimearmatures.html>

286 <http://www.users.bigpond.net.au/blendage/pages/tutorials/tut6/maze.html>

287 http://www.users.bigpond.net.au/blendage/pages/tutorials/tut10/adding_traps.html

288 http://www.tutorialsforblender3d.com/Game_Engine/MouseLook/MouseLook_First_1.html

289 http://www.tutorialsforblender3d.com/Game_Engine/Viewports/Viewport_1.html

290 <http://www.tutorialsforblender3d.com/GameFunctions/GameLogicPage1.html>

291 http://www.tutorialsforblender3d.com/Game_Engine/Vehicle/Vehicle_1.html

292 http://en.wikibooks.org/wiki/Blender_3D%2FTutorials%2FStoryboard

293 <http://www.deadtoadkingdom.com/BlEnDeR.html>

294 <http://www.cgmasters.net/free-tutorials/python-scripting/>

- A whole series of tutorials by nilunder²⁹⁵.

187.15. Python and Plugins

- Python Lessons²⁹⁶ * exposed (Broken Link)
 - Make Hair²⁹⁷ * exposed (Broken Link)
 - Povanim Export Script²⁹⁸
 - AI Path Importer²⁹⁹
 - 3D-No Plugins; Put your 3D Blender space on web!³⁰⁰
 - Randomizer Script³⁰¹ Scroll down the page
 - Embedding the 3D web plugin in a web page³⁰² (Broken Link)
 - Different Useful Scripts³⁰³
 - AfterGlow, Polyline, Hitchcock ZoomEffect, ...³⁰⁴ (Broken Link)
 - Dynamica³⁰⁵ (Broken Link)
 - BlendSaber³⁰⁶ (Broken Link)
 - Lsystem tree maker³⁰⁷
 - Python Scripting for Procedural Animation³⁰⁸
 - Focalblur - A Matter of Depth³⁰⁹ (Broken Link)
 - Python API Introduction³¹⁰
 - Python API, Making a Square Mesh³¹¹
 - Python API, Iterations³¹²
 - Python API, Automating Vertex Creation³¹³
 - Python API, Automating Face Creation³¹⁴
 - Python API, Making Potatoid³¹⁵
 - Python Script, To build an Empty for EnvMap ³¹⁶

295 <http://www.nilunder.com/tutorials>

296 http://www.sutabi.tk/timmeh/index.php?option=com_content&task=category§ionid=4&id=20&Itemid=26

297 <http://www.dedalo-3d.com/index.php?filename=SXCOL/makehair/abstract.html>

298 http://jmsoler.free.fr/util/blenderfile/fr/povanim_en.htm

299 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_import_ai_en.htm

300 <http://www.thoro.de/portfolio/verschiedenes/3DNP.html>

301 <http://www.elysiun.com/forum/viewtopic.php?t=37400>

302 http://www.blender3d.org/Education/index_old.php?sub=TutorialEmbedplugin

303 <http://www.selleri.org/Blender/scripts/text.html>

304 <http://www.hgb-leipzig.de/~daniel/blender/>

305 <http://www.centraresource.com/blender/dynamica/>

306 <http://www.blenderwars.com/tut.php?module=blendersaber>

307 http://marief.soler.free.fr/Monsite/lssystem_en.htm

308 <http://www.ingiebee.com/Blendermania/Genos%20Spring.htm>

309 <http://www.elysiun.com/tutorials.php?id=2>

310 http://jmsoler.free.fr/didacticiel/blender/tutor/english/python_script00.htm

311 http://jmsoler.free.fr/didacticiel/blender/tutor/english/python_script01.htm

312 http://jmsoler.free.fr/didacticiel/blender/tutor/english/python_script02.htm

313 http://jmsoler.free.fr/didacticiel/blender/tutor/english/python_script03.htm

314 http://jmsoler.free.fr/didacticiel/blender/tutor/english/python_script04.htm

315 http://jmsoler.free.fr/didacticiel/blender/tutor/english/python_script05.htm

316 http://jmsoler.free.fr/didacticiel/blender/tutor/english/eng_scriptmirror.htm

- Python Script, Bezier Curves Import³¹⁷
- Python Script, Paths import³¹⁸
- Python Script, Importing Adobe Illustrator Format³¹⁹
- Python Script, Lionhead The Movies .msh import/export³²⁰
- Python Script, Mesh Explosion³²¹
- Python Script, Level Of Detail³²²
- Python Script, Wire Shadows and extrusions³²³
- Python Script, Changing the active camera instantly³²⁴
- Subsurface Scattering in Blender³²⁵ (Broken Link)
- Using the Superficial Scattering Script³²⁶
- City Block Generator³²⁷
- Blender Camera Calibration with Live Camera³²⁸
- Import-export different formats, Different Generators, etc...³²⁹ (Broken Link)
- Horn Extrude³³⁰ (Broken Link)
- Mesh shaker and tutorial³³¹
- Batch STL³³²
- Vertices to a Curve Converter³³³
- Zutils, Z-Buffer Utilities³³⁴
- Setting Vertex Colors with Python³³⁵

187.16. Using other Programs with Blender

- YafRay³³⁶
- Verse³³⁷ (Broken Link)
- SPE - Python IDE for Blender³³⁸

317 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_curvesimport_en.htm

318 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_paths_import_en.htm

319 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_import_ai_en.htm

320 http://www.dcmoothing.com/main/index.php?option=com_docman&task=cat_view&Itemid=&gid=36&orderby=dmdatecounter&ascdesc=DESC

321 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_meshexplosion_en.htm

322 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_LOD_en.htm

323 http://jmsoler.free.fr/didacticiel/blender/tutor/python_wireshadows_en.htm

324 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_changerdecamera_en.htm

325 http://www.dedalo-3d.com/index.php?filename=SXCOL/experiments/ss_scattering_python.html

326 <http://feeblemind.tuxfamily.org/dotclear/index.php/2005/04/25/39-mh-tutorial-using-the-superficial-scattering-script>

327 <http://www.elysiun.com/forum/viewtopic.php?t=16217>

328 <http://www.elysiun.com/forum/viewtopic.php?t=32715>

329 <http://www.redrival.com/scorpius/blender-plugins.htm>

330 <http://www3.sympatico.ca/emilio.aguirre/hornextrude.html>

331 <http://saltshaker.sourceforge.net/>

332 http://blender.formworks.co.nz/index.pl?p=2z/batch_stl.html

333 <http://www.elysiun.com/forum/viewtopic.php?t=45576>

334 <http://www.elysiun.com/forum/viewtopic.php?t=35355>

335 <http://www.blender3dclub.com/index.php?name=News&file=article&sid=54>

336 <http://www.yafRay.org>

337 <http://blender.org/modules/verse/>

338 <http://pythonide.stani.be>

- Wings 3D Subdivision Modeler³³⁹
- Verse Gimp-Blender Plugin³⁴⁰
- Kerkythea Renderer³⁴¹
- Equinox 3D³⁴²
- Wings3D, a Quick and Accurate UVmapping Tool for Blender³⁴³ (Broken Link)
- Povray export³⁴⁴
- Adobe Illustrator Paths import³⁴⁵
- Inkscape SVG import³⁴⁶
- Using Blender content in PowerPoint 2000³⁴⁷ (Broken Link)
- Ter2Blend³⁴⁸ (Broken Link)
- Batch Processing Images³⁴⁹
- BVH Info³⁵⁰
- Stereogram³⁵¹ (Broken Link)
- Importing VRML³⁵² (Broken Link)
- Creating Quicktime VRs³⁵³
- Blender, Python and Mac OS X³⁵⁴
- Stylistic Rendering³⁵⁵ (Broken Link)
- Voodoo Camera Tracker³⁵⁶
- Lionhead The Movies³⁵⁷
- Using Blender with IMVU³⁵⁸ IMVU - Information³⁵⁹ (Broken Link)

187.16.1. Distributed Computing

- BURP - Big and Ugly Rendering Project³⁶⁰

339 <http://www.wings3d.com/>

340 <http://users.pandora.be/blendix/verse/old/demo.html>

341 <http://www.kerkythea.net/joomla/>

342 <http://www.equinox3d.com>

343 <http://mywebpages.comcast.net/jmandmc/uvmap/uvmap.html>

344 http://jmsoler.free.fr/util/blenderfile/fr/povanim_en.htm

345 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_import_ai_en.htm

346 http://jmsoler.free.fr/didacticiel/blender/tutor/cpl_import_svg_en.htm

347 http://www.blender3d.org/Education/index_old.php?sub=TutorialPowerpoint

348 <http://users.skynet.be/sky33676/ter2blend1.html>

349 http://membres.lycos.fr/bobois/Tuts/traitement_par_lot/traitement_par_lot_ang.html

350 <http://www.centraresource.com/blender/bvh/>

351 <http://www.geocities.com/blenderlab/tutorial1.html>

352 http://www.blender3d.org/Education/index_old.php?sub=TutorialVrml

353 http://www.katorlegaz.com/index.php?filename=/articles/blender/Quicktime_VR.php

354 http://www.katorlegaz.com/index.php?filename=/articles/blender/Python_and_OS_X.php

355 <http://www.flippyneck.com/temp/NPR.htm>

356 <http://www.digilab.uni-hannover.de/docs/manual.html>

357 http://www.dcmadding.com/main/index.php?option=com_smf&Itemid=79&topic=180.0

358 <http://neo5842.starlithost.info/my%20video%20tuts%2001.html>

359 http://www.imvu.com/catalog/web_landing.php?p=power&userId=10209797

360 <http://burp.boinc.dk/>

- Global Rendering-Farm³⁶¹
- OS X Distributed Blender Network Rendering with Xgrid³⁶² (Incorrect link?)
- Bfarm Distributed Rendering via Internet³⁶³ (Broken Link)
- Powua³⁶⁴ Blender renderfarm Requires a Password

187.16.2. Maybe someday ...

- LightRay³⁶⁵ (Broken Link)
- Toxic³⁶⁶

187.17. Video Tutorials

- Blender Underground³⁶⁷
- Official Blender Video Tutorials³⁶⁸ (Broken Link)
- Super3boy's Blender Tutorials³⁶⁹
- Greybeard's Blender Video Tutorials³⁷⁰
- Series of short VTuts that explain basic details³⁷¹
- lessons on Blender 3d³⁷² (Broken Link)
- Japanese Blender2.59 Video Tutorials³⁷³

187.18. Blender WikiBooks

- Blender 3D: Blending Into Python³⁷⁴
- Blender 3D: HotKeys³⁷⁵
- Blender 3D: Import and Render a SolidWorks Model³⁷⁶
- Blender 3D: MemoBook³⁷⁷ * exposed

361 <http://renderworld.futureware.at/>

362 http://www.katorlegaz.com/index.php?filename=/articles/blender/Using_Blender_with_Xgrid.php

363 <http://geocities.com/tronovan3d/>

364 <http://www.powua.com/>

365 <http://www.tacc.utexas.edu/~cburns/lightray/lightray.php>

366 <http://www.toxicengine.org/>

367 <http://blenderunderground.com/video-tutorial-list/>

368 <http://www.blender.org/tutorials-help/video-tutorials/>

369 http://www.youtube.com/profile_videos?user=super3boy

370 <http://www.ibiblio.org/bvidtute/>

371 <http://www.blendernation.com/2006/03/01/videotutorials-blender-3d-workshop/>

372 <http://www.animationforcreation.comFree>

373 <http://orionstaff.blog84.fc2.com/blog-entry-1562.html>

374 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Blending%20Into%20Python>

375 <http://en.wikibooks.org/wiki/Blender%203D%3A%20HotKeys>

376 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Import%20and%20Render%20a%20SolidWorks%20Model>

377 <http://en.wikibooks.org/wiki/Blender%203D%3A%20MemoBook>

- [Blender 3D: Noob to Pro](#)³⁷⁸
- [All Blender WikiBooks Modules](#)³⁷⁹

187.19. FAQ

- [Blender FAQ](#)³⁸⁰ (Generated on September 24, 2001)
- [Blender Tips](#)³⁸¹ (Broken Link)

187.20. Miscellaneous

CG artists *must see or read:*

- [How to succeed in Animation](#)³⁸²
- [Bitter History of 3D Business](#)³⁸³
- [Blender Art Gallery](#)³⁸⁴ (Broken Link)
- [Classical Film and Video Knowledge Base](#)³⁸⁵
- [Quick Tips in Design&GFX](#)³⁸⁶

187.20.1. Open Movies

- [Orange](#)³⁸⁷ (Broken Link)
- [NaNo - Blender Internet Virtual Movie Studio](#)³⁸⁸

187.20.2. IRC

- <irc://irc.freenode.net/blender>
- <irc://irc.freenode.net/blenderchat>
- <irc://irc.freenode.net/blenderclasses>
- <irc://irc.freenode.net/blenderqa>
- <irc://irc.freenode.net/blendercoders>
- <irc://irc.freenode.net/blenderwiki>
- <irc://irc.freenode.net/gameblenderdev>
- <irc://irc.freenode.net/gameblender>

378 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro>

379 <http://en.wikibooks.org/wiki/%3ACategory%3ABlender>

380 http://www.museum.state.il.us/ismdepts/library/linuxguides/blender/blender_faq_0.html

381 http://dev.newmediaworx.com/johnnyb/bplanet/tut_blendertips.htm

382 <http://genedeitch.awn.com/>

383 <http://home.nordwest.net/Adger/tips/tip077.html>

384 <http://centralsource.com/blenderart/index.php>

385 <http://www.rondexter.com/>

386 <http://www.atpm.com/9.07/design.shtml>

387 http://orange.blender.org/cms/The_Movie.555.0.html

388 <http://nano.prods.free.fr/>

- <irc://irc.freenode.net/verse>

187.20.3. Tests

- [Blender Benchmarks](#)³⁸⁹

187.21. Other Lists

Please read more about on [talk](#)³⁹⁰ page.

187.21.1. Tutorials

- [blenderartists.org](#)³⁹¹ * exposed
 - [Categorized list of Blender 3D Tutorials](#)³⁹²
 - [A Monolingual List of Tutorials](#)³⁹³ French Language
 - [A Multilingual List of Tutorials](#)³⁹⁴
 - [Dave Jarvis' Blender Tutorials](#)³⁹⁵
 - [Blender Tutorials/Ebooks in PDF format](#)³⁹⁶
 - [Dark Scarab Tutorials](#)³⁹⁷

187.21.2. Miscellaneous

- [Blender3d Club : Tutorials, samples, Links.](#)³⁹⁸
- [SurfnLearn.com : Blender 3D Tutorials](#)³⁹⁹
- [Blender 3D Links & Resources](#)⁴⁰⁰
- [Blender Heads around the Globe\(Where live other Blenders?\)](#)⁴⁰¹ (Broken Link)
- [Blenderart Magazine](#)⁴⁰² (Broken Link) [Mirror 1](#)⁴⁰³ (Broken Link), [M2](#)⁴⁰⁴ (Broken Link), [M3](#)⁴⁰⁵ (Broken Link)

389 <http://www.eofw.org/index.php>

390 <http://en.wikibooks.org/wiki/Talk%3ABlender%203D%3A%20Tutorial%20Links%20List>

391 <http://blenderartists.org/cms/index.php?id=38>

392 http://www.surfnlearn.com/view_tutorials/Blender_3D/All/

393 <http://www.zoo-logique.org/3D.Blender/index.php3?zoo=dif>

394 http://membres.lycos.fr/bobois/Liens_Links/big_list.html

395 <http://www.davidjarvis.ca/blender/>

396 <http://blendercourse.com/English.aspx>

397 <http://www.darksscarab.com/tutorials>

398 <http://www.blender3dclub.com>

399 http://www.surfnlearn.com/view_tutorials/Blender_3D/All/

400 http://www.katorlegaz.com/index.php?filename=links/blender_3D.php

401 <http://www.frappr.com/blenderheadsaroundtheglobe>

402 <http://s12.yousendit.com/d.aspx?id=14ZVYHLTFHOAK2V3YTCWU38RQ3>

403 http://www.apollux-designs.com/BA-Magazine/blenderart_mag-nov-05.pdf.zip

404 <http://blendertestbuilds.de/index.php?dir=Blenderart/20051115/>

405 http://www.intellidesign.org/blenderart_mag/

- Blender Classroom Tutorial⁴⁰⁶
- Blender Tutorials Search Engine⁴⁰⁷ (Broken Link)
- Blender Basics 2nd ed - excellent classroom course⁴⁰⁸
- BlenderMasters⁴⁰⁹ - Many resources for blender users
- **LANGUAGES (czech)** ⁴¹⁰

187.22. About

This links list is language filtered and extended version of personal collection originally provided by **IamInnocent**. So if you looking for tutorials in other languages check this link at www.eYsiun.com⁴¹¹.

- German Blenders should have a look at http://de.wikibooks.org/wiki/Blender_3D:_Tutorial_-_Linkliste

If you want to add Blender tutorial link in some other language you can add it temporary on talk⁴¹² page. We will later made such WikiBooks in other languages too.

Thank you all who contributed to this nice and useful links collection! Feel free to add your name or link if you think you need to be mentioned here.

de:Blender 3D: Tutorial Linkliste⁴¹³

406 <http://www.statikonline.com/Blender/>

407 <http://www.Blender3DTutorials.com>

408 <http://www.cdschools.org/54223045235521/blank/browse.asp?A=383&BMDRN=2000&BCOB=0&C=55205>

409 <http://www.blendermasters.com>

410 <http://www.youtube.com/watch?v=fsHW7J9dTNE>

411 <http://www.elysiun.com/forum/viewtopic.php?t=13380>

412 <http://en.wikibooks.org/wiki/Talk%3ABlender%203D%3A%20Tutorial%20Links%20List>

413 <http://de.wikibooks.org/wiki/Blender%203D%3A%20Tutorial%20Linkliste>

188. Hotkeys

Blender's user interface is very flexible and often provides more than one way to do something. Many operations that can be done by clicking on buttons and menus can also be done with keyboard shortcuts called "hotkeys". The hotkey is usually much faster than the corresponding mouse action, so it's a good idea to learn them as soon as possible.

An entire WikiBook (titled *Blender 3D: HotKeys¹*) is devoted to Blender hotkeys. We encourage you to refer to it as often as necessary.

¹ <http://en.wikibooks.org/wiki/Blender%203D%3A%20HotKeys>

189. Output Formats

Blender can render still images and video to many different file formats. Here is an alphabetical list:

- **AVI Codec** - AVI codec compression. Available codecs are operating system dependent. When an AVI codec is initially chosen, the codec dialog is automatically launched. The codec can be changed directly using the Set Codec button in AVI Codec settings.
- **AVI Jpeg** - AVI but with Jpeg compression. Lossy, smaller files but not as small as you can get with a Codec compression algorithm. Jpeg compression is also the one used in the DV format used in many digital camcorders.
- **AVI Raw** - Audio-Video Interlaced (AVI) uncompressed frames.
- **BMP** - Bit-Mapped Paint is an early lossless format.
- **Cineon** - format produced by a Kodak Cineon camera and used in high-end graphics software and directed toward digital film.
- **DPX** - Digital Moving-Picture eXchange format; an open professional format (close to Cineon) with meta-information about the picture; 16-bit uncompressed bitmap (huge file size). Used in preservation.
- **FFMPEG** - Fast Forward Moving Pictures Expert Group is a collection of free and open source software libraries that record, convert and stream digital audio and video in numerous formats. You must have the proper codec installed on your computer for Blender to call it and use it to compress the video stream through FFMPEG, but there are preset formats to choose from, such as DV, SVCD, and DVD.
- **Frameserver** - Blender puts out frames upon request as part of a render farm. The port number is specified in the OpenGL User Preferences panel.
- **HamX** - Blender's own self-developed 8 bits RLE (Run Length Encoded bitmap) format; it creates extremely compact files that can be displayed quickly. To be used only for previsualization of animations (Play button).
- **Iris** - the standard Silicon Graphics Inc (SGI) format used on Unix OS machines.
- **JPEG** - Joint Photographic Experts Group (name of the consortium which defined it), an open format that supports very good compression with little loss of quality. Only saves RGB value. Re-saving images results in more and more compression and loss of quality.
- **JPEG 2000**
- **MultiLayer** - an OpenEXR format that supports storing multiple layers of images together in one file. Each layer stores a renderpass, such as shadow, specular, color, etc. You can specify the encoding used to save the MultiLayer file using the codec selector (ZIP (lossless) is shown and used by default).
- **OpenEXR** - an open and non-proprietary extended and highly dynamic range (HDR) image format, saving both Alpha and Z-depth buffer information.
- **PNG** - Portable Network Graphics, a lossless image format often used in web design.
- **QuickTime** - A proprietary multimedia framework for video and images.
- **Radiance HDR**
- **Targa**

- **Targa Raw**
- **TIFF** - Tagged Image File Format created by Aldus for desktop publishing.

If the selection seems daunting, here are pointers:

If you need transparency in your images (i.e., an alpha channel that allows any part of the rendered image to have varying degrees of opacity), or you will be compositing, you generally want to use:

- Multilayer
- PNG
- OpenEXR
- Targa

Of the above four formats, you will find that PNG and Targa images can be opened and edited by most any bitmap editing application, such as GIMP or Photoshop. However, OpenEXR and Multilayer offer more options within Blender, and may produce smaller, lossless files.

190. Image Portfolio

190.1. Summary

This is the Blender 3D: Noob to Pro¹ image portfolio page, where all the categories of the images used in the book are here in one place for your reference.

190.2. Categories

Blender 3D²

Blender 3D icons³

Blender 3D screenshots⁴

Non Blender 3D screenshots⁵

Blender 3D splash screens⁶

Blender 3D viewports⁷

Blender 3D suzanne primitive⁸

190.3. A sampling of images from the above categories

Heres a random sample of 200 images from the over 1,800 images in the above categories

-
- 1 <http://en.wikibooks.org/wiki/Blender%203D%3A%20Noob%20to%20Pro>
 - 2 <http://en.commons.org/wiki/Category%3ABlender%203D>
 - 3 <http://en.commons.org/wiki/Category%3ABlender%203D%20icons>
 - 4 <http://en.commons.org/wiki/Category%3ABlender%203D%20screenshots>
 - 5 <http://en.commons.org/wiki/Category%3ANon%20Blender%203D%20screenshots>
 - 6 <http://en.commons.org/wiki/Category%3ABlender%203D%20splash%20screens>
 - 7 <http://en.commons.org/wiki/Category%3ABlender%203D%20viewports>
 - 8 <http://en.commons.org/wiki/Category%3ABlender%203D%20suzanne%20primitive>

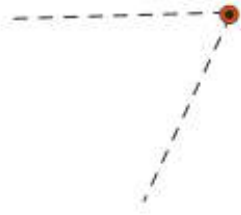


Figure 1117

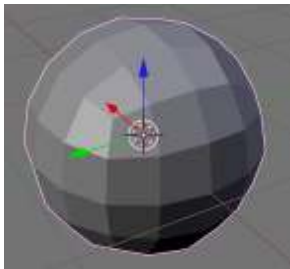


Figure 1118



Figure 1119



Figure 1120



Figure 1121



Figure 1122

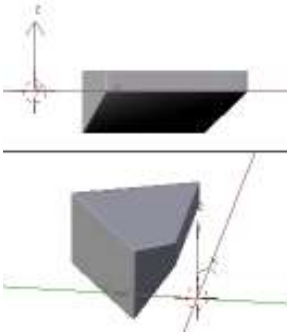


Figure 1123

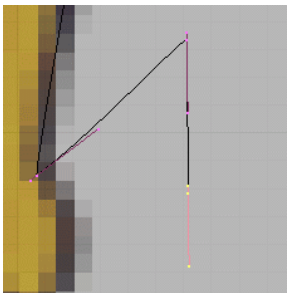


Figure 1124



Figure 1125

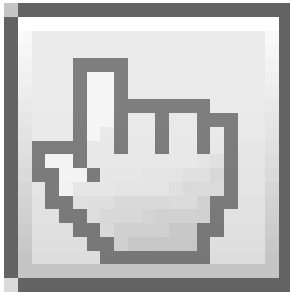


Figure 1126



Figure 1127



Figure 1128



Figure 1129

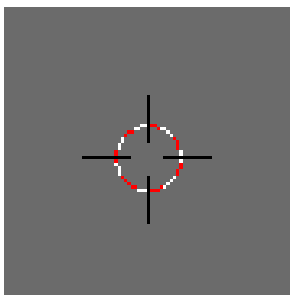


Figure 1130

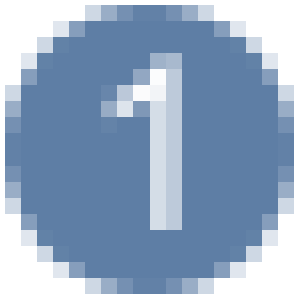


Figure 1131

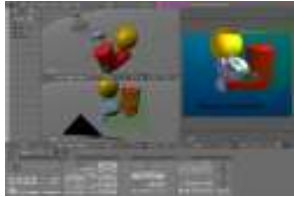


Figure 1132

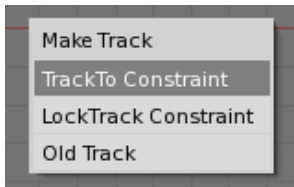


Figure 1133



Figure 1134



Figure 1135



Figure 1136



Figure 1137

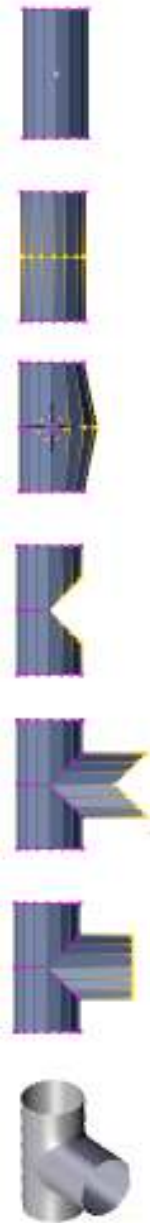


Figure 1138

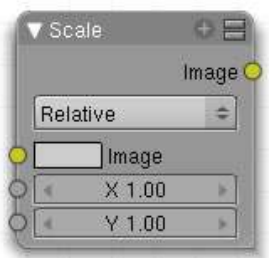


Figure 1139



Figure 1140



Figure 1141

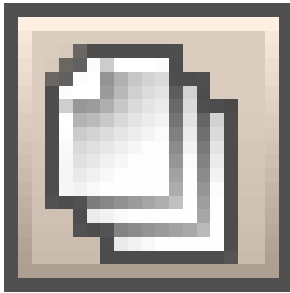


Figure 1142



Figure 1143



Figure 1144



Figure 1145



Figure 1146



Figure 1147



Figure 1148

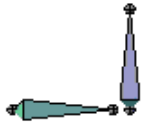


Figure 1149



Figure 1150

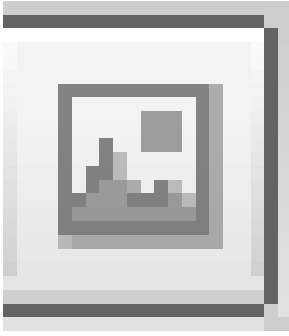


Figure 1151



Figure 1152

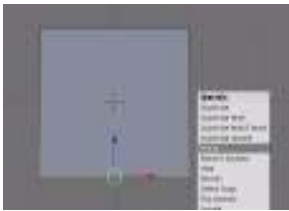


Figure 1153



Figure 1154



Figure 1155

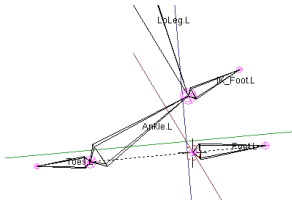


Figure 1156



Figure 1157

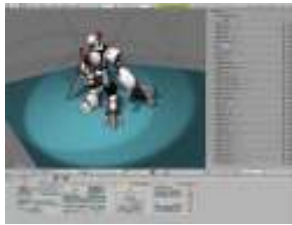


Figure 1158

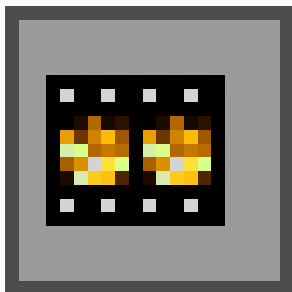


Figure 1159

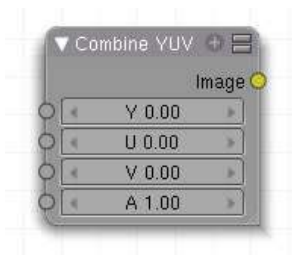


Figure 1160



Figure 1161

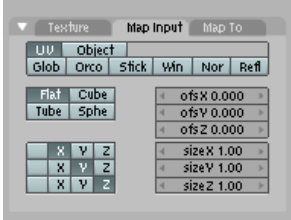


Figure 1162

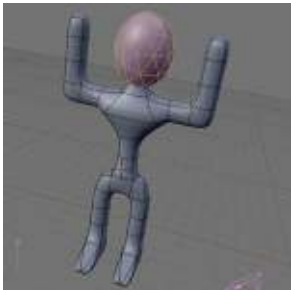


Figure 1163



Figure 1164



Figure 1165

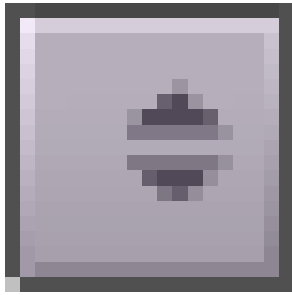


Figure 1166

Figure 1167

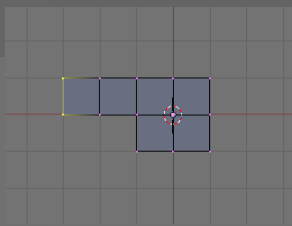


Figure 1168



Figure 1169



Figure 1170

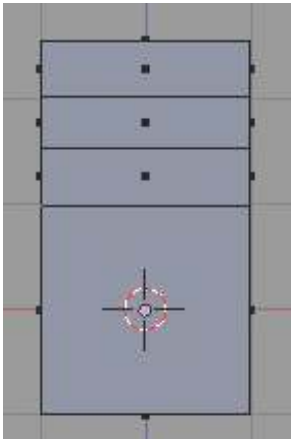


Figure 1171



Figure 1172



Figure 1173

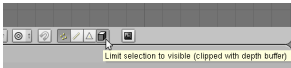


Figure 1174



Figure 1175

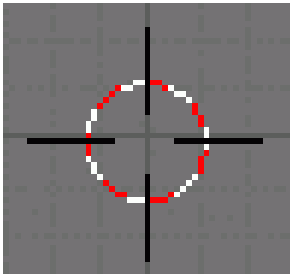


Figure 1176

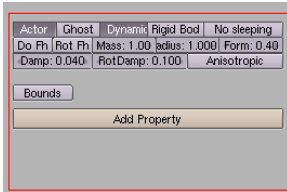


Figure 1177



Figure 1178



Figure 1179



Figure 1180



Figure 1181

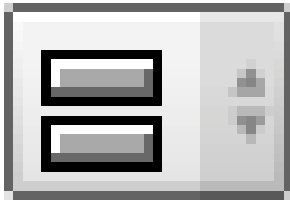


Figure 1182

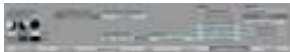


Figure 1183

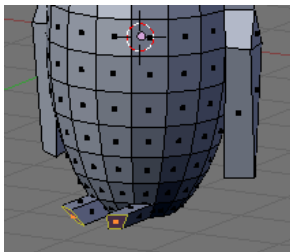


Figure 1184

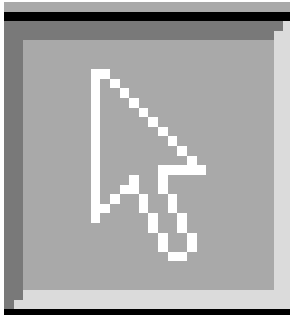


Figure 1185



Figure 1186



Figure 1187



Figure 1188

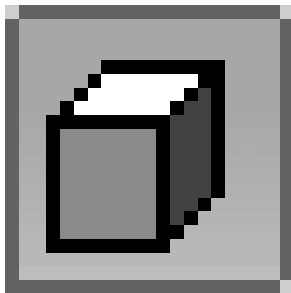


Figure 1189



Figure 1190



Figure 1191



Figure 1192

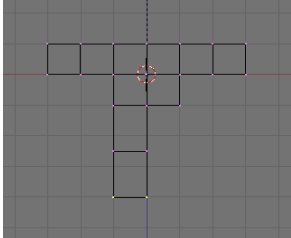


Figure 1193



Figure 1194

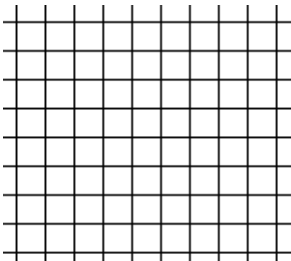


Figure 1195



Figure 1196

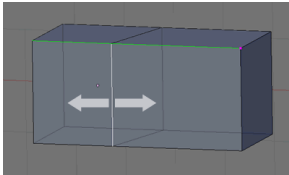


Figure 1197

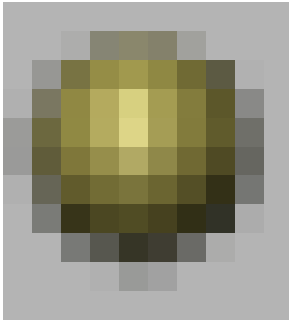


Figure 1198

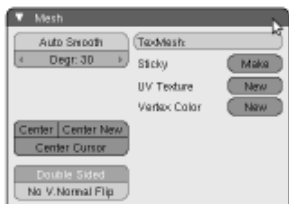


Figure 1199



Figure 1200



Figure 1201



Figure 1202



Figure 1203



Figure 1204



Figure 1205



Figure 1206

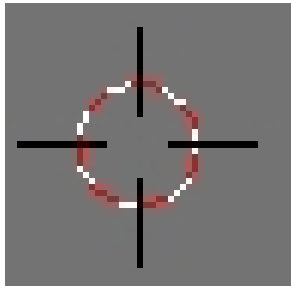


Figure 1207

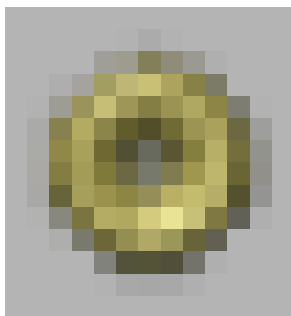


Figure 1208

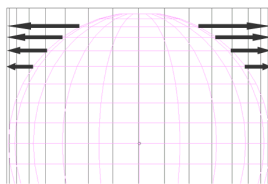


Figure 1209



Figure 1210

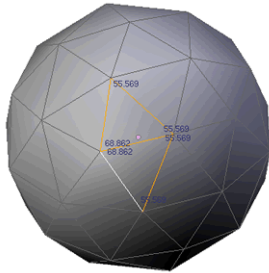


Figure 1211

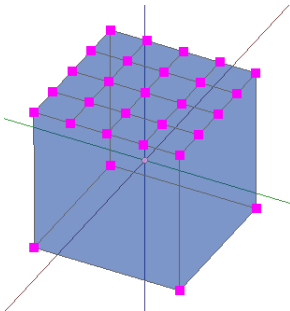


Figure 1212

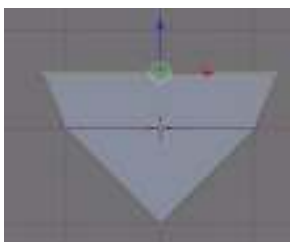


Figure 1213



Figure 1214



Figure 1215



Figure 1216



Figure 1217



Figure 1218



Figure 1219



Figure 1220



Figure 1221

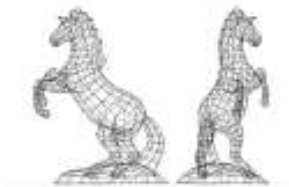


Figure 1222



Figure 1223



Figure 1224

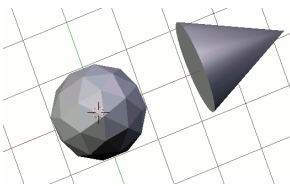


Figure 1225



Figure 1226

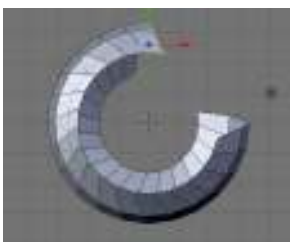


Figure 1227

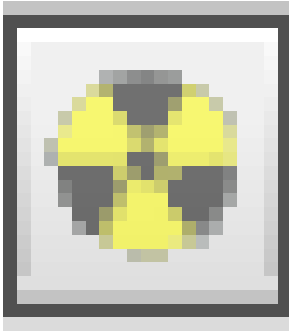


Figure 1228

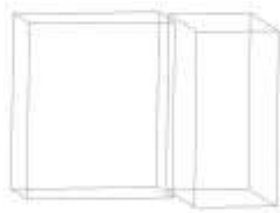


Figure 1229

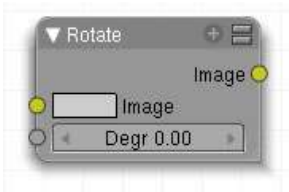


Figure 1230

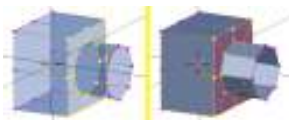


Figure 1231

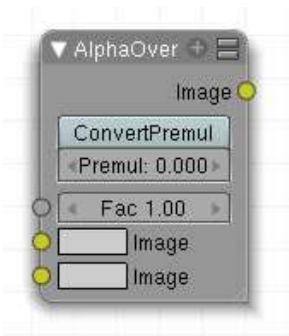


Figure 1232



Figure 1233

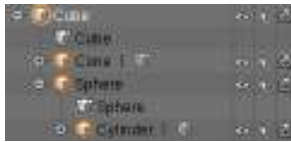


Figure 1234

Lock X Rot	Lock Y Rot	Lock Z Rot
Set X: 0.000		Set Z: 0.000
Unit X:		Unit Z:
Min X: -94.7		Min Z: -64.6
Max X: 116.4		Max Z: 66.6

Figure 1235



Figure 1236

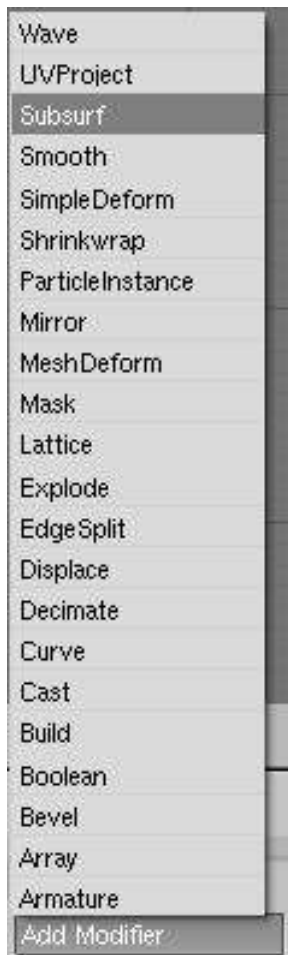


Figure 1237

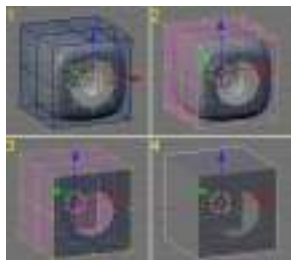


Figure 1238



Figure 1239

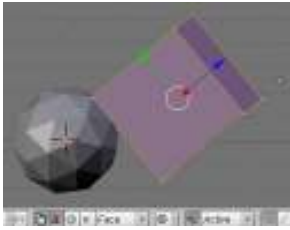


Figure 1240

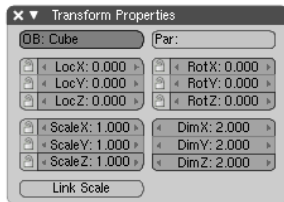


Figure 1241

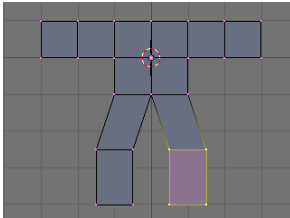


Figure 1242

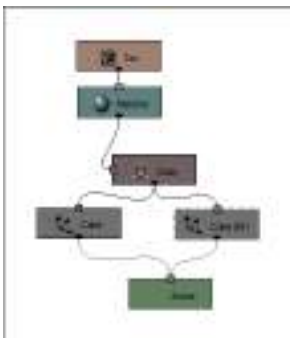


Figure 1243



Figure 1244

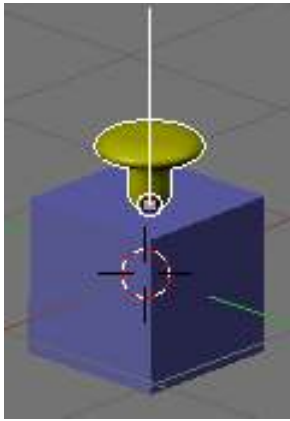


Figure 1245



Figure 1246



Figure 1247

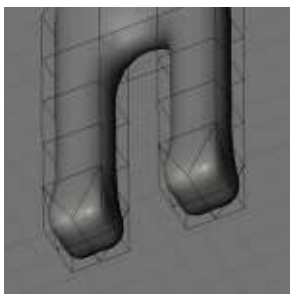


Figure 1248



Figure 1249

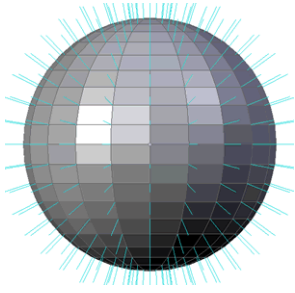


Figure 1250

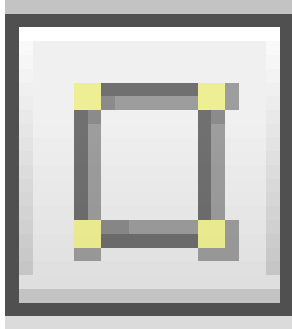


Figure 1251



Figure 1252

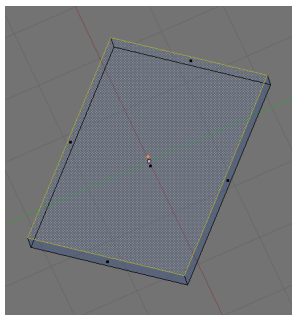


Figure 1253

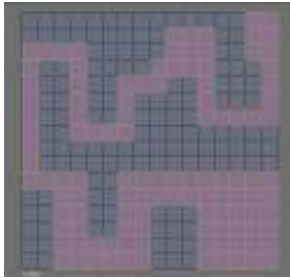


Figure 1254



Figure 1255

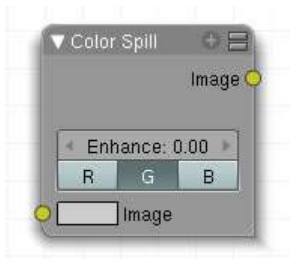


Figure 1256



Figure 1257

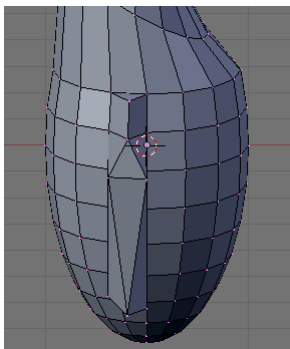


Figure 1258

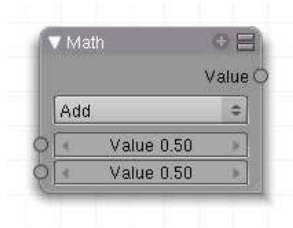


Figure 1259



Figure 1260

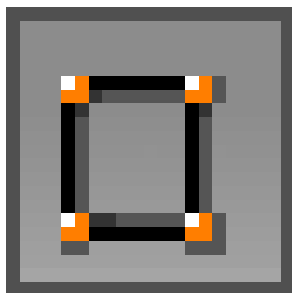


Figure 1261



Figure 1262

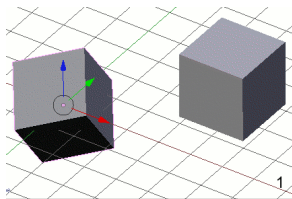


Figure 1263

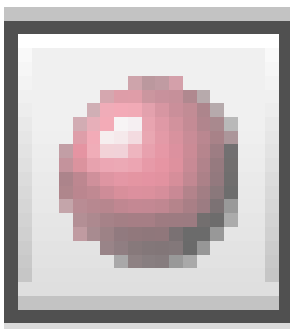


Figure 1264



Figure 1265

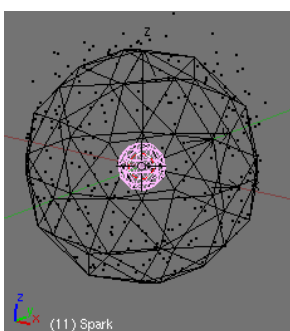


Figure 1266



Figure 1267

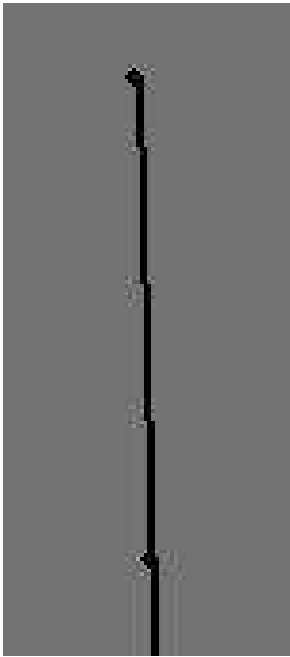


Figure 1268

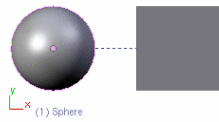


Figure 1269



Figure 1270

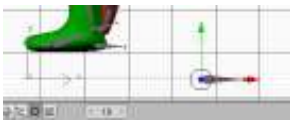


Figure 1271

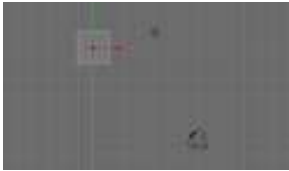


Figure 1272



Figure 1273



Figure 1274

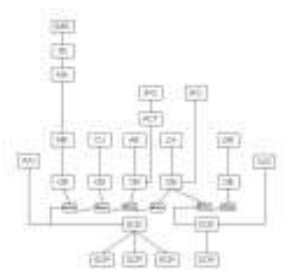


Figure 1275



Figure 1276



Figure 1277

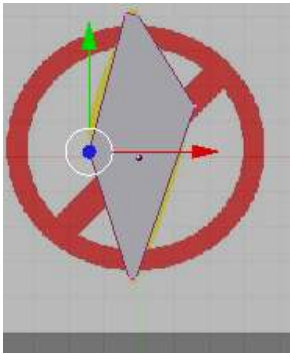


Figure 1278



Figure 1279



Figure 1280

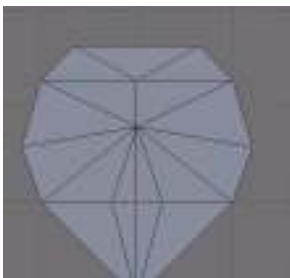


Figure 1281

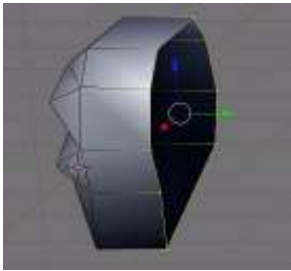


Figure 1282



Figure 1283



Figure 1284



Figure 1285

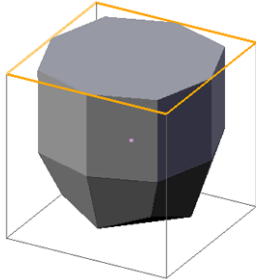


Figure 1286

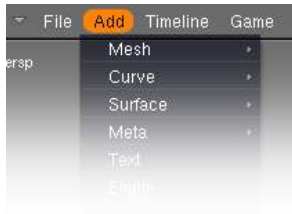


Figure 1287

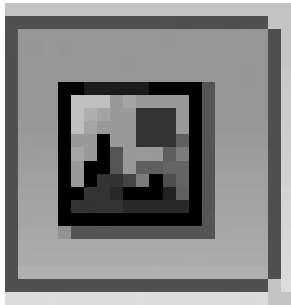


Figure 1288



Figure 1289

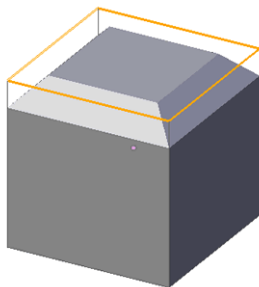


Figure 1290

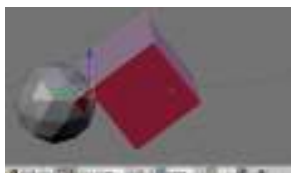


Figure 1291

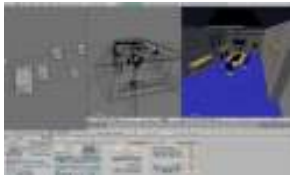


Figure 1292

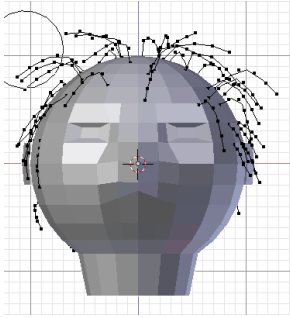


Figure 1293



Figure 1294

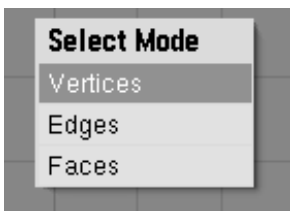


Figure 1295

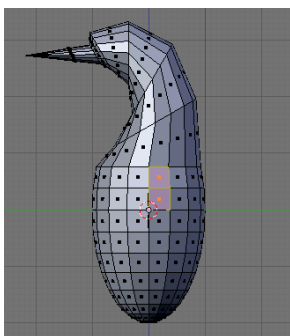


Figure 1296

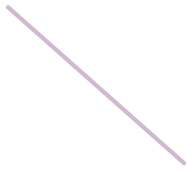


Figure 1297



Figure 1298

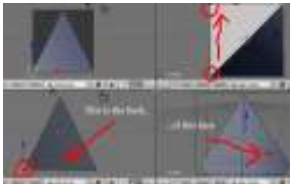


Figure 1299

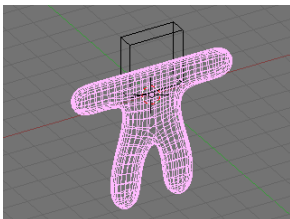


Figure 1300



Figure 1301



Figure 1302

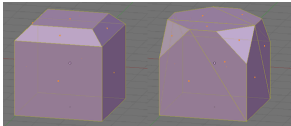


Figure 1303



Figure 1304



Figure 1305

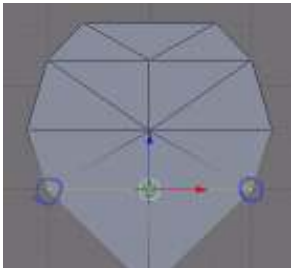


Figure 1306

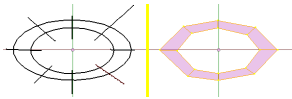


Figure 1307



Figure 1308

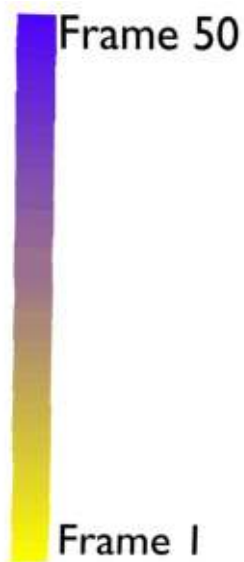


Figure 1309



Figure 1310

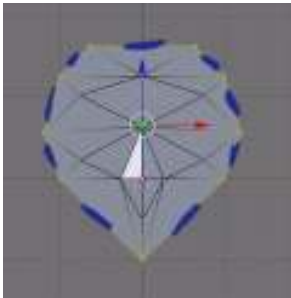


Figure 1311

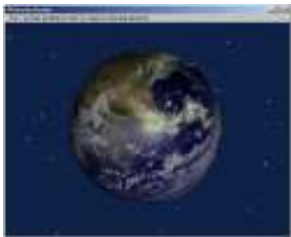


Figure 1312

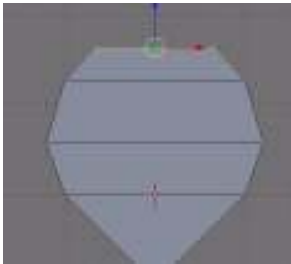


Figure 1313

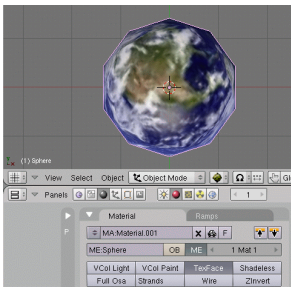


Figure 1314

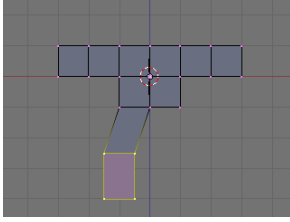


Figure 1315

Image Portfolio⁹

⁹ <http://en.wikibooks.org/wiki/Category%3ABlender%203D%3A%20Noob%20to%20Pro%2FProject>

191. Blender Glossary

Please add words and phrases to the alphabetical glossary at Blender 3D: Noob to Pro/Glossary¹.

191.0.1. General Terminology

- **2D** (adjective): two-dimensional (flat)
- **3D** (adjective): three-dimensional
- **axis** (noun): one of the primary directions used to define a coordinate system; plural: **axes**
- **coordinates** (noun): a string of numbers (usually two or three) which denote a location
- **coordinate system**: a scheme for denoting locations by means of coordinates
- **disappearing taskbar**: a taskbar which appears only when the mouse pointer reaches an edge of the display
- **geometry**:
 1. the branch of mathematics concerned with positions, directions, angles, lengths, areas, volumes, and so on
 2. the portion of a 3D model that deals with shapes
- **GIMP**: an open-source software tool for editing 2D computer graphics
- **illusion of depth**: the human propensity to imagine a 3D scene when presented with a flat (2D) picture of it
- **light** (verb): to describe the intensity, color, direction, etc. of the light sources in a 3D scene
- **model** (noun): an abstract description of a 3D object (or scene) which is suitable for rendering
- **model** (verb): to create models
- **numpad**: a numeric keypad
- **orthographic view**: a 2D view which represents a scene's parallel edges with parallel edges; contrast with: perspective view
- **perpendicular** (adjective): forming a 90-degree (right) angle
- **perspective view**: a 2D view which represents a scene's parallel edges with edges that converge toward a vanishing point
- **pose** (verb): to position objects and their parts in a 3D scene
- **render** (verb): to generate a 2D view from a 3D model of a scene, by means of a computer
- **rig** (noun): to describe how different parts of an object are capable of moving relative to one another
- **texture** (verb): to describe how a 3D object interacts with light (scattering, bending, etc.)

191.0.2. Blender User Interface Terminology

- **active window**: the window currently responding to keystrokes

¹ Chapter 185 on page 1327

- **blender unit**: the height and width of each grid square in a 3D View window
- **header**: a row of controls along the top or bottom of a window
- **panel**: a rectangular area in a Buttons window which can be rearranged relative to other panels
- **window** (noun): a rectangular area of Blender's user-interface which can be split or resized
- **window type** (noun): one of the sixteen varieties of windows in Blender's user interface:
 - **Scripts Browser** - This window is used to initiate any Python² scripts saved in Blender to add functionality.
 - **File Browser** - Browses the files on your computer in Blender.
 - **Image Browser** - Browses the files on your computer in Blender with visuals of images displayed.
 - **Node Editor** - The enhanced material editing system in Blender in which you can customize the input/output of materials or scenes and composite them together.
 - **Buttons Window** - Displays the Blender scene and object editing options.
 - **Outlines** - Breaks down the entire collection of objects and their properties in an organized list.
 - **User Preferences** - Allows you to change Blender's controls and preferences.
 - **Text Editor** - Simple text editor for any notes/Python code.
 - **Audio Window** - Allows you to upload, enhance, and save audio files into the Blender library.
 - **Timeline** - Shows the animation's beginning, ending, and current frame, as well as a timeline with marks indicating all the key positions of an object.
 - **Video Sequence Editor** - Feature for the composition of sound, scenes, videos and other cinematographic elements.
 - **UV/Image Editor** - Allows you to add and alter image files in Blender and also apply these images onto the faces of meshes in an editable fashion as a material.
 - **NLA Editor** - Allows you to change the key positions of all actions/IPOs.
 - **Action Editor** - Allows you to alter the key positions of an armature and it's collection of bones.
 - **IPO Curve Editor** - Allows you to insert saved key points of the positions, transformations, and settings of objects of the course of a series of frames.
 - **3D View** - Window through which you work with, add, and place objects in a 3D world.

191.0.3. Blender Modeling Terminology

- **Quad *n*** - A face with four sides.
- **Mesh *n*** - A collection of faces, edges, and vertices which can be modeled and manipulated in Edit Mode.
- **Vertex (*pl. vertices*) *n*** - A 3-dimensional coordinate, which in groups comprise a polygon. With default settings, it is represented in Blender by a purple dot when unselected, or a yellow dot when selected.
- **Edge *n*** - A wire-like line representing the boundary of 2 adjacent vertices.
- **Face *n*** - A planar connection of edges representing a boundary, field, or solid surface. Identified individually in Edit Mode as a black box in the middle of the corresponding face.
- **Icosphere *n*** - A sphere composed of *n* triangular faces composed in a manner to give the best "smoothness" for the lowest memory consumption.
- **UVSphere *n*** - A sphere composed of *n* square faces arranged in rings to allow the smoothest application of real-time movement of vertices/faces and application of images onto the spectrum of faces.

² <http://en.wikibooks.org/wiki/Python>

- **Lamps** - Category of 5 light-emitting objects:
 - Lamp - Light is emitted in the scene with a uniform spherical output which fades with distance, decreasing eventually to 0. The distance can be specified in its options.
 - Sun - Light is emitted in the scene with a uniform spherical output and remains uniform regardless of the distance from it.
 - Spot - Light is emitted in the scene in a predetermined conical area with light properties similar to lamp and also initially calculates the shadow (lack of light) present behind an object (Known as ray shadow).
 - Hemi - Sun-like light emitted in a 180 degree arc. Works just like a Sun lamp, but with less functionality. Best used for quick, broad illumination.
 - Area - Light which works similar to a Spot lamp but shape of the cone is definable.
- **Ray shadows n.** - Method of generating shadows cast by light by calculating the angle and faces of the interfering object to create a smooth, sharp and complete shadow. Memory-intensive, best quality.
- **Buffer Shadows n.** - Method of generating shadows cast by light by calculating the angle of a mesh's edge and creating a set amount of shadow "squares" on the shadow-receiving objects which are calculated from a set buffer-ratio. Memory-friendly, while possibly lower quality.
- **Camera n.** - An object which acts as the viewpoint (when active) for the scene when the scene is to be rendered.
- **Armature n.** - An object which contains 'bones'. Bones are sub-objects which are connected with each other in a chain, each bone-link is parented to the bone before it. The purpose is to parent any objects to the bone, and also you may parent a mesh to the entire armature and have selected vertices of the mesh parented to bone(s) percentage-wise (0% means no effect, 50% means vertices copy half the transformation value, 100% means vertices transformed as full child of bone.)
- **Empty n.** - [*Please add description here...*]

192. Every Material Known to Man

192.1. "Every Material Known To Man" Project

Want to share your material settings with the world? This is the place! This page is going to be turning from a page to a chapter, but to do that we need your support! If you have an idea for a material, even if you don't know how to make it, just put it in the list to help! Someone out there knows how to make it, and if they come across this page with it listed, they might be willing to share.

193. Sources of free 3D models

Here are some sources of free 3d models from across the web. Take them apart, put them back together, understand what makes them tick, to improve your own modeling.

193.1. Blender-Specific

These sites offer .blend files for download.

- The Official Blender Model Repository¹
- <http://www.blendswap.com/>² — specifically for swapping Blender models. But note that all Lego models are offered under the non-Free CC-BY-NC licence, due to some misunderstanding of how trademarks are supposed to work.
- Blender for Architecture³
- <http://opengameart.org/>⁴ — more Blender models
- http://katorlegaz.com/3d_models/index.php⁵ -- they have furniture, arcade video games, joysticks, toys, vehicles, road and sidewalk items, and simple human models
- Yorik's site⁶ — some Blender files for his architectural projects are available for download.

193.2. Non-Blender-Specific

These sites offer models in other formats.

- <http://archive3d.net/>⁷ - 15,000+ free models from archive 3d
- <http://sketchup.google.com/3dwarehouse/>⁸ - Google's 3d warehouse repository; take the collada version, when available (the Google Earth downloads are actually .zip files with a collada inside them, so just rename and unzip)

Note: some of the 3d warehouse models can't be understood by blender due to a zillion repeating elements, others will show up invisible (one sided, or inside out), be gigantic in size, or all parented to some crazy spot out in space. Fixing these to be usable will push your blender skills to new levels in some cases.

1 <http://e2-productions.com/repository/index.php>
2 <http://www.blendswap.com/>
3 <http://blender-archi.tuxfamily.org/Models>
4 <http://opengameart.org>
5 http://katorlegaz.com/3d_models/index.php
6 <http://yorik.uncreated.net/>
7 <http://archive3d.net>
8 <http://sketchup.google.com/3dwarehouse/>

Tips: I've tried importing a few of these 3D Warehouse models, and so far I've only noticed a couple of issues:

- The model imports into a new Blender scene, called something like “SketchupScene” or “SketchupScene001”. You may also find additional scenes created with nothing in them, that you can delete. If you try rendering, you may not see anything, because the default camera is only linked into the original default scene named “Scene”. Just select the camera and hit **CTRL+LKEY** to link it into the new scene.
 - You may notice parts of the model appearing and disappearing as you try viewing it from different angles. This is because it is scaled very large (several hundred Blender Units on a side). Try scaling it down so it's no more than a few BU on a side.
 - There will usually be a lot of doubles, so remove them with **WKEY**->remove doubles.
 - There may be many separate objects, so select what you want to combine and hit **CTRL+J** to join them all into one object.
- <http://artist-3d.com/>⁹ - wide variety of free models
 - http://www.nasa.gov/multimedia/3d_resources¹⁰ - all kinds of free models from the space program
 - <http://www.3dm3.com/modelsbank/>¹¹ - around 300 free models
 - <http://www.amazing3d.com/free/free.html>¹² - free section on amazing 3d about 75 models of all kinds of things.
 - http://www.corporatemedianews.com/2002/03_mar/features/3dmodellibrary.htm¹³ - free models from corporate media news; a very interesting collection from klingon space ships to footballs.
 - <http://3dplants.0catch.com/download.html>¹⁴ - about 70 free 3d plant models
 - <http://www.3delicious.net>¹⁵ - 3d delicious, lots of models, but way too many popover ads! (click the picture, then click the bigger preview in the next page to download)
 - <http://www.top3dmodels.com/products/3d-models/free>¹⁶ - 20 free models
 - <http://www.top3d.net/free-3d-models/>¹⁷ - about 85 models, including washing machines, tires, and jewelry
 - <http://www.oyonale.com/modeles.php?lang=en&format=OBJ>¹⁸ - about 17 free models form guns, to rubber duckies
 - <http://dmi3d.comeze.com/>¹⁹ - a few hundred free models of cars, race cars, trucks, tanks, backhoes, etc.
 - <http://www.scifi3d.com/list.asp?intGenreID=14&intCatID=36>²⁰ - free models of scfi empires like, Aliens, Babylon5, Blade Runner, BS Galactica, Original SciFi, Other SciFi, Star Trek, Star Wars
 - resources.blogscopia.com²¹ - models in Blender and other formats.

9 <http://artist-3d.com/>

10 http://www.nasa.gov/multimedia/3d_resources

11 <http://www.3dm3.com/modelsbank/>

12 <http://www.amazing3d.com/free/free.html>

13 http://www.corporatemedianews.com/2002/03_mar/features/3dmodellibrary.htm

14 <http://3dplants.0catch.com/download.html>

15 <http://www.3delicious.net>

16 <http://www.top3dmodels.com/products/3d-models/free>

17 <http://www.top3d.net/free-3d-models/>

18 <http://www.oyonale.com/modeles.php?lang=en&format=OBJ>

19 <http://dmi3d.comeze.com/>

20 <http://www.scifi3d.com/list.asp?intGenreID=14&intCatID=36>

21 <http://resources.blogscopia.com/category/models/>

- <http://www.hobbycncart.com/>²² - Free relief models (all models are free) in .STL .OBJ and depth map (.BMP) formats.

193.3. Sites Using .RAR Format

Separated out because they may not be decodable with Free Software.

- <http://www.3dmodelfree.com/>²³ - 3d model free, lots of variety here
- <http://dlegend.com/html/free-3dmodels.html>²⁴ - design company with a few hundred free 3d models
- <http://3ddb.blogspot.com/search/label/Models>²⁵ - lots of free luxury style models in sets (has a giant picture at the top, just scroll down).

193.4. Sites Requiring Registration

- http://www.turbosquid.com/Search/Index.cfm?keyword=&max_price=0²⁶ - 13,000+ free models at 3d stock site turbosquid (registration required for download)
- <http://www.3dextras.com/3dextras-free-3d-models-list.asp?catid=-1>²⁷ - 3d extras, lots of variety, but too many popover ads.
- <http://dd-freebies.blogspot.com/search/label/3D%20models>²⁸ - a few baubles and ornaments
- http://www.exchange3d.com/free-3d-models/cat_35.html?noredirect=1&act=viewCat&catId=35²⁹ - about 100 free models, some even made in blender!
- http://www.wirecase.com/Gallery-Free-3D-Models_s-2_v-20-1_f-fc-0-321_f-fpx-_f-fpn-_f-oid-.html³⁰ - 25 sets of models from airplanes to bottles
- <http://www.sharecg.com/b/5/3D-Models?PSID=e3aa02d1ef780c432e9f9d7e95409369>³¹ - wide assortment of free models
- <http://www.cg-files.com/3dmodels.html>³² - from tires to fans (lots of popover ads)
- <http://www.3dcontentcentral.com/default.aspx>³³ - Free 3D Models of Parts & Assemblies like brackets, resistors, motors, shafts, valves.

²² <http://www.hobbycncart.com/>

²³ <http://www.3dmodelfree.com/>

²⁴ <http://dlegend.com/html/free-3dmodels.html>

²⁵ <http://3ddb.blogspot.com/search/label/Models>

²⁶ http://www.turbosquid.com/Search/Index.cfm?keyword=&max_price=0

²⁷ <http://www.3dextras.com/3dextras-free-3d-models-list.asp?catid=-1>

²⁸ <http://dd-freebies.blogspot.com/search/label/3D%20models>

²⁹ http://www.exchange3d.com/free-3d-models/cat_35.html?noredirect=1&act=viewCat&catId=35

³⁰ http://www.wirecase.com/Gallery-Free-3D-Models_s-2_v-20-1_f-fc-0-321_f-fpx-_f-fpn-_f-oid-.html

³¹ <http://www.sharecg.com/b/5/3D-Models?PSID=e3aa02d1ef780c432e9f9d7e95409369>

³² <http://www.cg-files.com/3dmodels.html>

³³ <http://www.3dcontentcentral.com/default.aspx>

194. Blueprint Links List

This page contains links to 2D images and drawings that can be useful for constructing models.

- <http://www.shipschematics.net>¹ - Starship Schematics Database
- http://www.suurland.com/blueprints_archive.php
- <http://www.luft46.com/profiles/profiles.html>² - 3 views
- <http://www.aviastar.org/index2.html>³ - 3 views
- <http://www.rcgroups.com/forums/showthread.php?t=557457>⁴ - 3 views on a lot lot lot of airplanes
- http://www.google.com/intl/en/sketchup/3dwh/pdfs/modeling_a_city.pdf⁵ - Google's guide on how to model a city
- <http://www.boatdesign.net/boat-plans-archive/index.htm>⁶ boats
- <http://www.mgussin.freeuk.com/00Plans.htm>⁷ old cars
- http://www.airwar.ru/other/draw_1w.html⁸ War Planes
- <http://jpracing.racerplanet.com/>⁹ Cars, Trucks, Buses
- <http://www.swaqvalley.com/blueprints.php?section=samples>¹⁰ sports cars

194.1. Registration Required

- <http://www.the-blueprints.com/>¹¹ — requires registration for full-quality downloads
- <http://smcars.net/>

194.2. Payment Required

- <http://www.eplans.com/>¹² - floor plans, but if you navigate around a bit to find em.
- <http://www.emporis.com/application/>¹³ - heights and details of buildings; photos available for purchase

1 <http://www.shipschematics.net>

2 <http://www.luft46.com/profiles/profiles.html>

3 <http://www.aviastar.org/index2.html>

4 <http://www.rcgroups.com/forums/showthread.php?t=557457>

5 http://www.google.com/intl/en/sketchup/3dwh/pdfs/modeling_a_city.pdf

6 <http://www.boatdesign.net/boat-plans-archive/index.htm>

7 <http://www.mgussin.freeuk.com/00Plans.htm>

8 http://www.airwar.ru/other/draw_1w.html

9 <http://jpracing.racerplanet.com/>

10 <http://www.swaqvalley.com/blueprints.php?section=samples>

11 <http://www.the-blueprints.com/>

12 <http://www.eplans.com/>

13 <http://www.emporis.com/application/>

- <http://skyscraperpage.com/>¹⁴ - building heights, diagrams, but downloads and screenshots are prohibited

194.3. Shouldn't Really Be Here?

- http://www.archiplanet.org/wiki/Main_Page¹⁵ - building encyclopedia, can't find any actual plans

¹⁴ <http://skyscraperpage.com/>

¹⁵ http://www.archiplanet.org/wiki/Main_Page

195. Materials, Textures, Photos

Here are some sources of materials, textures and photos from across the Web.

195.0.1. Materials

- Large Free Repository of Materials¹

195.0.2. Textures

- Textures Library²
- Mayang's Free Textures Hi-Res³
- Jeremy Engleman's Public Textures Hi-Res⁴ (doesn't seem to display properly)
- Image*After Free Images Hi-Res⁵
- Free Textures Mid-Res⁶
- Sky Maps⁷

195.0.3. Photos

- <http://www.med.harvard.edu/AANLIB/cases/caseNA/pb9.htm>⁸ - Brain scan
- <http://www.morguefile.com>⁹ - the morgue file
- <http://www.sxc.hu>¹⁰ - Stock exchange
- <http://www.everystockphoto.com>¹¹ everystockphoto
- http://www.pixelperfectdigital.com/free_stock_photos¹² - Pixel Perfect Digital
- <http://openphoto.net>¹³ - OpenPhoto.Net
- <http://www.imageafter.com>¹⁴ - Image*After

1 <http://matrep.parastudios.de/index.php>

2 <http://textures.z7server.com/>

3 <http://www.mayang.com/textures/index.htm>

4 http://www.art.net/~jeremy/photo/public_texture_frameset.html

5 <http://www.imageafter.com/>

6 <http://digitalcraftsman.com/textureBin/textureBin.htm>

7 <http://www.elysiun.com/forum/viewtopic.php?t=24738>

8 <http://www.med.harvard.edu/AANLIB/cases/caseNA/pb9.htm>

9 <http://www.morguefile.com>

10 <http://www.sxc.hu>

11 <http://www.everystockphoto.com>

12 http://www.pixelperfectdigital.com/free_stock_photos

13 <http://openphoto.net>

14 <http://www.imageafter.com>

- <http://www.aboutpixel.de>¹⁵ - aboutpixel.de
- <http://www.freephotoweb.com/freephotoweb>¹⁶ - Free Photos Web
- <http://www.bigfoto.com>¹⁷ - Big Foto
- <http://www.freephotobank.com>¹⁸ - FreePhotosBank.com
- <http://www.freejpg.com.ar>¹⁹ - FreeJPG
- <http://www.stockvault.net>²⁰ - Stockvault.net
- <http://www.freepixels.com>²¹ - FreePixels
- <http://www.freerangestock.com>²² - Freerange Stock

15 <http://www.aboutpixel.de>
16 <http://www.freephotoweb.com/freephotoweb>
17 <http://www.bigfoto.com>
18 <http://www.freephotobank.com>
19 <http://www.freejpg.com.ar>
20 <http://www.stockvault.net>
21 <http://www.freepixels.com>
22 <http://www.freerangestock.com>

196. Asking for Help

So you've come to learn Blender, eh? You've made a great choice. Blender is one of the most powerful 3D animation and 3D creation tools out there, especially if you're short on cash. Learning how to use Blender can be a daunting task, so **don't give up!** With the help of this wikibook, you can become a Blender power user and put those Maya folks to shame.

If you ever get stuck in a tutorial for some reason, there are a number of places you can turn for help. The best way to get help is with an Internet Relay Chat (IRC) client such as X-Chat¹. Connect to **irc.freenode.net** and talk to blender users in the following channels.

- [irc://irc.freenode.net/blenderwiki #blenderwiki]
- [irc://irc.freenode.net/blender #blender]
- [irc://irc.freenode.net/blenderchat #blenderchat]
- [irc://irc.freenode.net/blenderqa #blenderqa]
- [irc://irc.freenode.net/gameblender #gameblender]

If you can't get help there, click the "discussion" tab at the top of the page that you're having trouble with, and explain your problem on that page. Wait at least 24 hours for help.

If you're still not getting help, try asking for help in the BlenderArtists.org forums².

es:Blender 3D: novato a profesional/Clases particulares de principiante³

1 <http://xchat.org>

2 <http://blenderartists.org/forum>

3 <http://es.wikibooks.org/wiki/Blender%203D%3A%20novato%20a%20profesional%20Clases%20particulares%20de%20principiante>

197. Know What You're Doing

The following tips may improve your chances of enjoying a successful project with Blender.

197.0.4. Organize early, organize thoroughly

In Blender projects, you may work with complex scenes containing hundreds of items: models, lights, cameras, scripts, materials, and more. This can be overwhelming if it is not properly organized.

We suggest that you organize as you go, giving each item a descriptive name. There will be ways to search for items by name, so start early. Similarly, give each file a descriptive name and organize files into folders (or directories): a separate folder for each project, with sub-folders for renders, texture images, background images, and so on.

197.0.5. Don't bite off more than you can chew

One of the greatest challenges faced by Blender users is their own imagination. While it's great to be creative and imaginative, you have to pace yourself. Be aware of your limits and the limits of your computer.

When making something from scratch, start with a very simple version of it and add details gradually.

Test frequently; see how well your recent changes work before proceeding to the next step. In this way, you will meet with fewer problems, and the ones you do run into will be simpler and easier to fix because there will be fewer possible causes to consider.

Monitor the complexity of your projects, watching for "lag" or excessive disk activity which might mean you are nearing your computer's limits.

When working on the computer, take frequent breaks to rest your eyes and stretch your muscles. Eat, sleep and exercise regularly. While focused time at the computer is necessary, inspiration and solutions to problems often come when our bodies are relaxed and our mental focus is elsewhere. While you may be tempted to go wild, keeping a level head while swimming in your imagination will give you a much more satisfying result with fewer headaches along the way.

198. Modeling Realistically

Everyone who starts a new career or hobby in 3D Modeling also seems to look around their immediate environment for things to model. Things like your computer monitor, your keyboard, and even the desktop which they are resting are all common "first models." Developing an eye for geometry takes time and this is a great chance to practice.

For this tips and techniques discussion, I'll start with the concepts of furniture and walls, but the ideas really extend to anything and everything you can model in Blender, from spaceships to fantasy demon-women, from your barbecue deck blueprints to biologically accurate squids.

198.1. Assemble Things, Don't Chisel Them

Let's start with the furniture around you.

You may think it's easy to model your desk using a couple of boolean operations on cubes. But it rarely comes out looking quite right. It seems like such a good idea at first. Then you find that the drawers won't open realistically. The corners are all amazingly sharp and crisp. The writing surface seems to be at the wrong scale for thickness. The woodgrain goes in the same direction everywhere. You then realize it will be hard to bevel just the few edges that need beveling. The more you look at it, even with good render settings, the more it looks like dollhouse furniture instead of a real life, full size desk.

Real furniture is not chiseled out of a half-ton cube of mahogany, because trees don't grow that thick, and it's a real waste to chip out the spaces for drawers. Instead, cabinetmakers tend to assemble a piece of furniture from many component parts. Think of those "self-assembly" kits from the local superstore, or even the showroom examples from a nearby unfinished knotty pine store.

Model several kinds of boards and fittings, then copy and assemble them appropriately. You don't need to model every nail and dowel, and you don't even need to know what a dovetail joint¹ is. But you can make realistic furniture when you build things from their natural components.

- Cut lumber in a variety of sizes and thicknesses.
- Measure real example furniture and take note of the construction details.
- Choose varying woodgrains by duplicating materials and adjusting them to be more individual.
- Align those woodgrains or other materials in an appropriate direction and scale for the model.
- Bevel just the parts or edges which are naturally worn or intentionally beveled.
- Any cloth, glass or metal fittings are separate components using their own materials.
- Create an "Empty" type object, give it a descriptive name, and parent all of the pieces to it.

There are a couple more benefits to this form of component modeling:

¹ <http://en.wikipedia.org/wiki/dovetail%20joint>

- Working with things made from multiple real materials is easier: one object, one material.
- The model is more flexible, because they can be reconfigured to other styles more easily.
- Components come in handy for other models: adapt a desk's drawers in kitchen cabinets and bedroom furniture.

198.2. Separate Objects for each Material



Figure 1316 skateboards, modeled in parts

A common question for new modelers is how to make one mesh that uses more than one material. For example, a skateboard with metal trucks, rubber tires, and a wooden deck. While Blender does provide an interface to allow such a combination, it is not usually the best approach for modeling realistically.

If you model the skateboard as separate parts, based on the way different real parts are assembled, then you have a lot more control over the final model. You can replace the wheels or use trucks of different styles later. You can adjust the mesh that defines the shape of the deck, without having to re-assign the areas that should be one material or the other. And lastly, you can choose to use subsurface tricks on the various parts, while avoiding them on other parts. Even if the whole model needs subsurfaces, it becomes a bit hard to control where one material starts, and another material ends.

198.3. Removing the Fourth Wall

Just as with furniture, you may model your first building interior as a hollowed-out cube. It seems straightforward to make a big cube, turn it inside out, and call it a living room. There are a lot of drawbacks to working with a room that was carved out of a single chunk of solid rock.

Buildings and rooms are assembled from parts. Just like the furniture modeling tip, design your rooms as a collection of separate pieces.

- You don't need to measure out the internal beams and joists, but remember they're there.
- Each interior or exterior wall should be its own object or assembly.
- Take care to note the correct dimensions and placement of doorways, windows, ceiling.

The benefits of a component-modeled interior include:

- Easier to see inside, just hide some walls on a different layer.
- Walls are much easier to move around to reconfigure the room layout.
- The same modeling work on one area can be copied to other rooms more easily.
- A more realistic architecture with consistent scaling is achieved quickly.

198.4. Proportions and Measurements

For realism, it's vitally important to get the right proportions for familiar objects. If your audience sees a bookshelf with 5" thick shelves, or a human hand which has a 5" long thumb, their brain is going to tell them that something is wrong. They might not be able to decide why, but they'll tell you it looks fake.

Even when you're animating fantasy characters, you should develop an understanding for proportion. Many cartoon characters have very large heads when compared to their bodies; this allows room for greater expression and makes the characters seem younger. Deciding how many "heads high" each character should appear will ensure that your figures all maintain proper proportions to each other and to their environment.



Figure 1317 soda cans and glass, modeled in parts

To get good proportions, you need to think about measurements.

- Grab a ruler or find some existing blueprints.
- Think in terms of those measurements for the duration of the project.
- Choose a base scale that will let you work comfortably within a 500x500x500 unit volume.
- Blender limits cameras to 1000 unit distances, for better numerical accuracy.

You can mix and match base units, by using Empty objects as parents. For example, you can model your furniture in inches or centimeters, and model your house in feet or meters. Then when you append your bed.blend's objects, just parent that bed to a new Empty object, and scale the Empty. This will conveniently scale all of the separate component pieces of the object together.

There's only one big wrinkle with working with multiple scales: materials should be designed with a certain base scale in mind. Many material surface properties are scaled according to the units you choose for a given object, or the world units. Just consider this as you design, and check your work critically.

199. Modeling tips

This is not really a tutorial, but a list of handy tricks you will probably use a lot when modeling.

1: Extrusion: to extrude, press EKEY in edit mode with the vertices selected, move and click to place the extrusion. you'll basically need this for every model in blender, without it, most models would be really hard to create

2: Set Smooth: setting smooth will smoothen everything, without increasing the amount of polygons and vertices you can use it for models with a smooth surface. to apply set smooth, select the mesh, go to the editing buttons (F9) and in the Links and Materials tab, press Set Smooth. if you press the Set Solid button, it will go back to normal.

3: Remove Double: when you remove double vertices, they will melt together, this is good for attaching things to each other. to remove double vertices, select the vertices, go to editing (F9) in edit mode, and in the Mesh tools tab, press Rem Doubl, the limit button next to it is the distance that the vertices have to have to be melted together.

Example use of these techniques:

Extrusion: <http://www.geocities.com/gauravnawani/tutorials/beginner/a-vase-modeling.html>¹ (they explain subsurf here too, but I'd recommend set smooth, because it doesnt increase the number of polygons and subsurf does) http://biorust.com/index.php?page=tutorial_detail&tutid=77&lang=en

Remove Double: http://biorust.com/index.php?page=tutorial_detail&tutid=77&lang=en² (same tutorial as in the extrusion section)

Set smooth: <http://www.blender.org/documentation/html/x2681.html>

If you want to add more handy techniques, please add them.

1 <http://www.geocities.com/gauravnawani/tutorials/beginner/a-vase-modeling.html>

2 http://biorust.com/index.php?page=tutorial_detail&tutid=77&lang=en

200. Cheat the 3D

200.1. Ways to Improve Performance

In 3D Graphic Design, there are many issues to consider. First, consider how the project moves, how the project performs. Next, consider how you create objects in your project, especially when working with meshes and lighting.

200.1.1. Mesh

Modifying an object's properties

When you add a modifying property to an object, the computer has to calculate that property every time it moves in the animation. These modifying properties include soft bodies, particles, or mirror textures. They can significantly slow down the computer.

To make things move faster, pre-calculate and "bake" the modifier into the object. To bake a modifier, such as a soft body movement, save the modifier as a permanent animation. That way no matter how the object changes, it moves the same way. As a result, your computer does not calculate how the object moves. The move is already set.

There is a drawback. If the object moves and hits another section with different slopes, it will continue to move as if it was in the original baking site. You will have to calculate and bake the section again to get new results.

Making 2D backgrounds

Scenes will have a foreground, background, sky, and more.

Your computer will take more time to render a scene that is big, open, and filled with modeled objects. This wastes time and memory, especially when modeling an object far in the distance. I'm guessing however, you do not want to make a quick, plain model. So what do you do?

That is easy. You make a 2D background.

- Put the distant image or background on a plane.
- Verify that everything on the plane around the image is black.

If it is not black, set the Alpha to 0 if the image was saved in a format that supports Alpha. This will retain the detail but reduce the number of faces the computer must count in the scene.

- Set the image to Billboard to prevent the image from skewing by perspective.

200.1.2. Materials

Using UV maps

For those who do not know the term, a UV map is an image applied to a 2D plane. The 2D plane is then applied to the mesh.

Here is how to create a UV map:

- Spread out the image onto a 2D plane.
- Apply the 2D plane over the face of the mesh using the 3D view.

201. Performance vs. Quality

201.0.3. Mesh

Polygon Count - when you work with meshes in 3D, be aware that the memory usage in Blender with reference to meshes comes from the number of faces (or vertices) the mesh contains. That means that if you have many useless faces (such as 55 faces on the side of a mesh that are as flat as one face), the number of faces will directly affect the Render time of Images and Animations. Also remember that for smooth rounded meshes, you do not need an infinite number of faces to make it look really nice and smooth. You actually don't need that many faces to keep a smooth slope looking smooth when viewed from the side. The viewer's eye will not actually identify every flat face, but will trace it as a smooth curve.

Face Structure - 4-side (quad) vs.3-side (tri) - Poly Count is very important when working in 3D graphics, and depending on the type of mesh you are making you could be wasting a vast number of faces because they are not the right structure. For example, if you want a smooth object, you can make it look mostly smooth when viewed from the side if you make it with well-structured 3-vertex faces. You will also yield the same results with fewer faces than if you use 4-vertex faces. Another example is that if you need to apply UVTexture, or have something to animate, a mesh being evenly deformed will yield a lower face-count if it is a 4-vertex versus a 3-vertex face-structure.

201.0.4. Lighting

Using RayShadows or Buffer Shadows - There are a few options you have for using shadows. Ray Shadows use an advanced algorithm to trace the edge of any interfering objects and create a perfect shadow onto the receiving object(s); however, the Ray Shadow calculation is memory-intensive and can seriously slow down your Render-Time. Buffer Shadows, on the other hand, use a different algorithm for similar results. The difference is that buffer shadows use a bit-rate of shadow "pixels" that fill in the shadowing area. You can adjust the bit-rate to make the shadow higher or lower quality. This calculation is much more memory-friendly, and your Render-Time will not jump up as much as with Ray Shadows. An interesting topic because this will change with better hardware performance and eventually won't matter. Until we get into 3D.

202. Modeling a Gingerbread Man

In this tutorial you will learn how to make a simple gingerbread man. In a later tutorial you will be able to make an animation with this gingerbread man.

In this tutorial we will tie together everything we've talked about up to this point, including extruding, subdividing and rendering, and throw in basic lighting.

202.1. Modeling

First, start Blender. You should see the usual top view of a cube in the 3D window, surrounded by a grid 'floor'.

Review— Zoom in or out with **SCROLL** or **CTRL+MMB**. Pan with **SHIFT+SCROLL** and **CTRL+SCROLL**. Make sure you are in orthographic view. Press **NUM5** to toggle between orthographic view and perspective view. You can tell you're looking at an orthographic view while looking down from the top (**NUM7**) of the cube: wherever you pan the window, you never see any part of the cube except the top.

- Display editing controls in the buttons window by clicking on the Editing context button

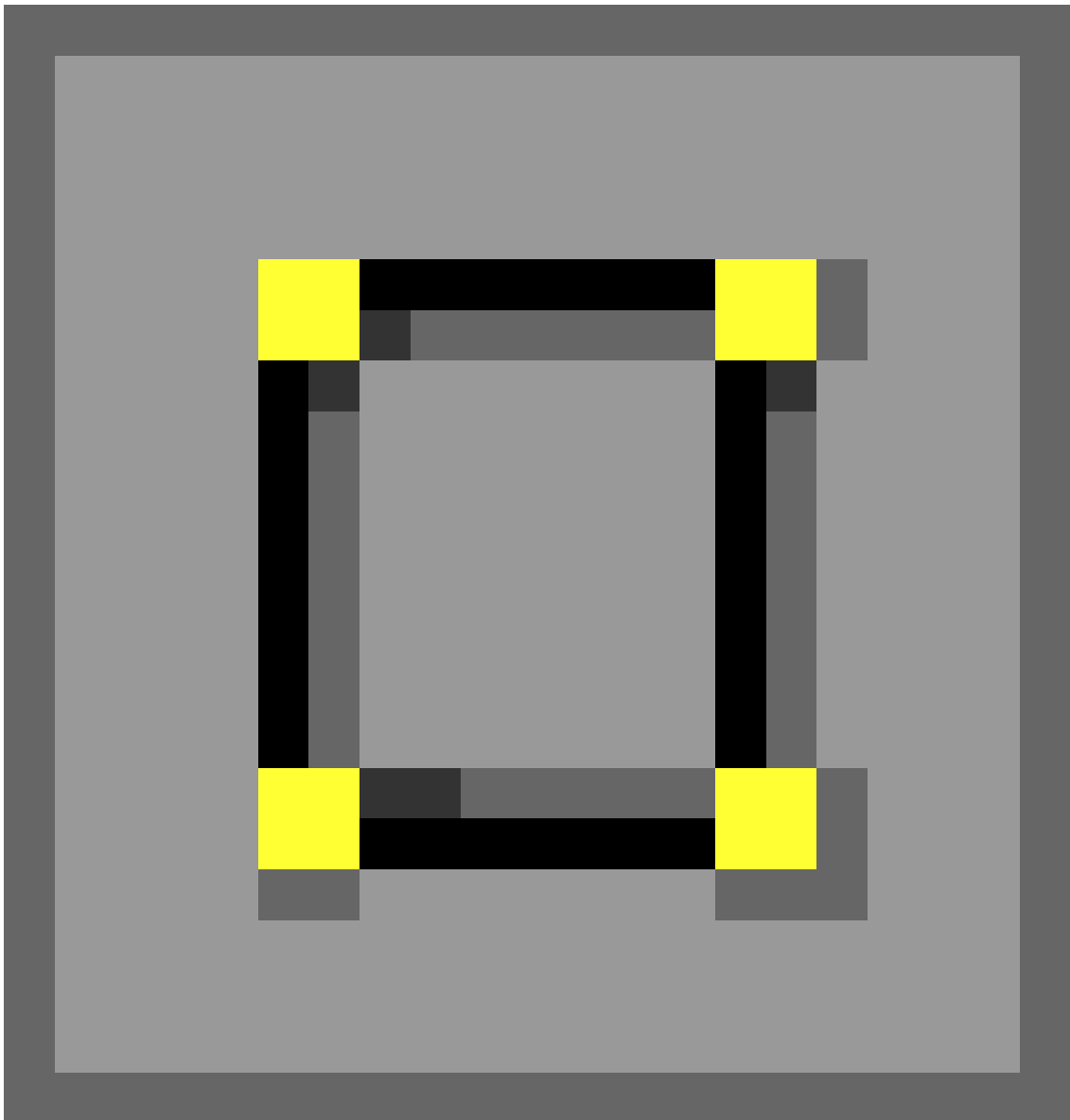


Figure 1318

or by pressing **F9**.

202.1.1. Adding Vertices

- Select the cube by clicking **RMB** on it. (It may already be selected.) Remember, selected objects are outlined in pink.
- Switch from object mode to edit mode by pressing **TAB**, which toggles between object and edit modes. In edit mode, at first, you'll see colored dots at the vertices of the cube. Selected vertices are highlighted in yellow. Unselected vertices are pink. (You'll see the dots when editing in vertex select mode. In edge select or face select mode, you'll see colored edges and faces instead.)

- Select all vertices of the cube. Press **AKEY** once. If the vertices are yellow, you've selected all of them. If they are pink, you've deselected all of them. If necessary, press **AKEY** again so that all vertices are selected/yellow.

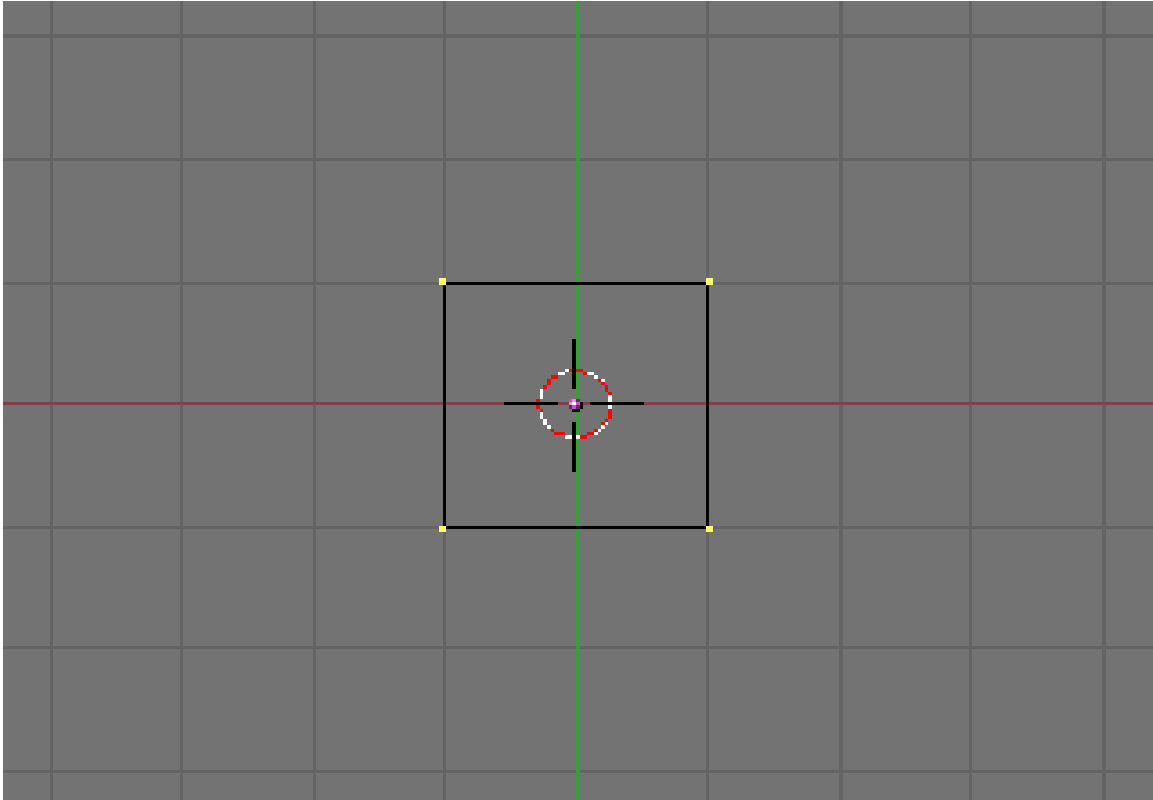


Figure 1319

- Subdivide the faces of the cube, using any of the following methods. (All vertices of the cube should still be selected.)
 - Click the Subdivide button in the Mesh Tools panel in the buttons window.
 - With the mouse pointer in the 3D window –
 - Press **WKEY** to display the Specials menu, then choose Subdivide.
 - Press **SPACE** to display the toolbox, then choose Edit → Edges → Subdivide.
 - Click the 3D window's Mesh menu, and choose Edges → Subdivide.
- Your cube now has more vertices. **Subdividing** edges adds vertices so you can create more complex shapes.

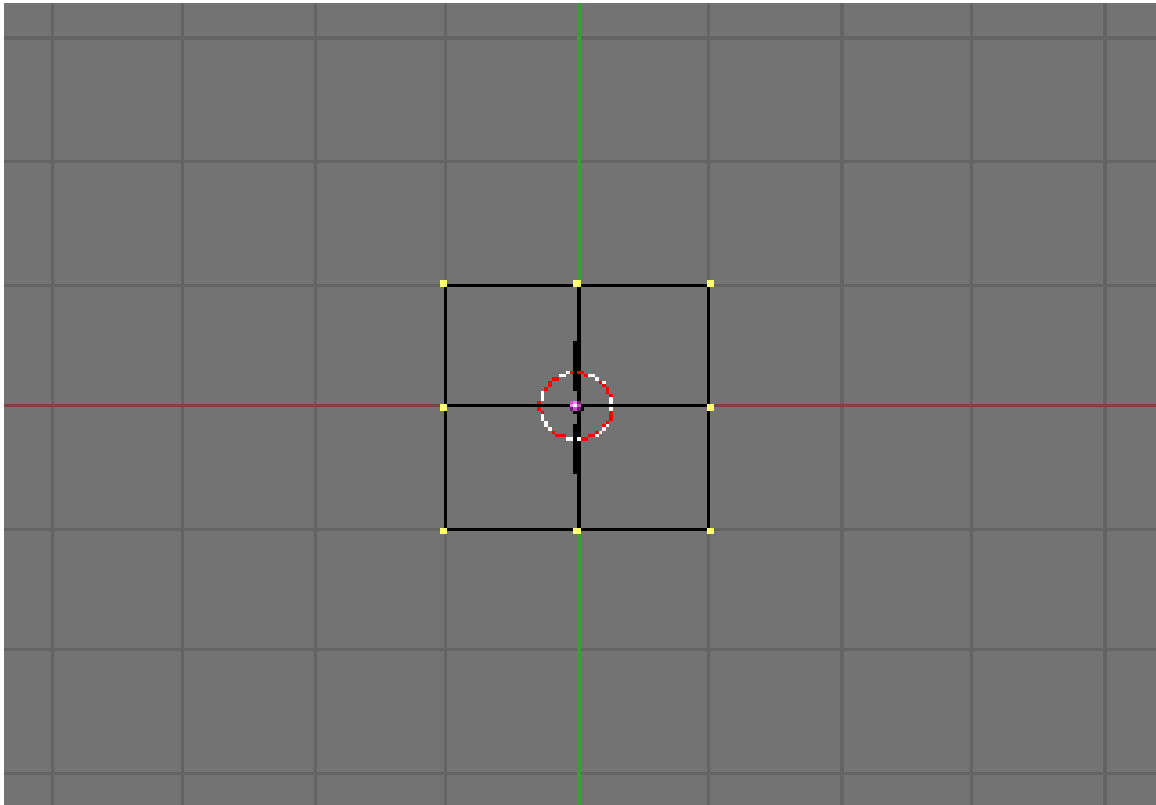


Figure 1320

202.1.2. Selecting a Subset of Vertices

We'll select six vertices on the upper half of one side of our cube, where we'll extrude an arm.

- Deselect all vertices by pressing **AKEY**.
- View the cube from the front (**NUM1**).
- Select the six vertices on the top half of the left side of the cube (see below) using one of the following methods.
 - Press **BKEY** and drag a rectangle around the top left and middle left vertices (viewing from the front).
 - Press **BKEY** twice to see a circle around your mouse pointer. **SCROLL** to change the size of the circle. Position the circle around the top left and middle left vertices, together or one at a time, and click **LMB**. Click **RMB** to finish.
- Take a closer look at the selected vertices by viewing the model from a different angle (drag with **MMB**).
- If you find that you have only selected two vertices and not six, make sure the "Occlude Background Geometry" button is off. That button is the right-most of the selection mode buttons, below. (It's called "Limit selection to visible" in Blender 2.45 and earlier. It doesn't appear if Blender is drawing in wireframe style.) Try selecting the vertices again.

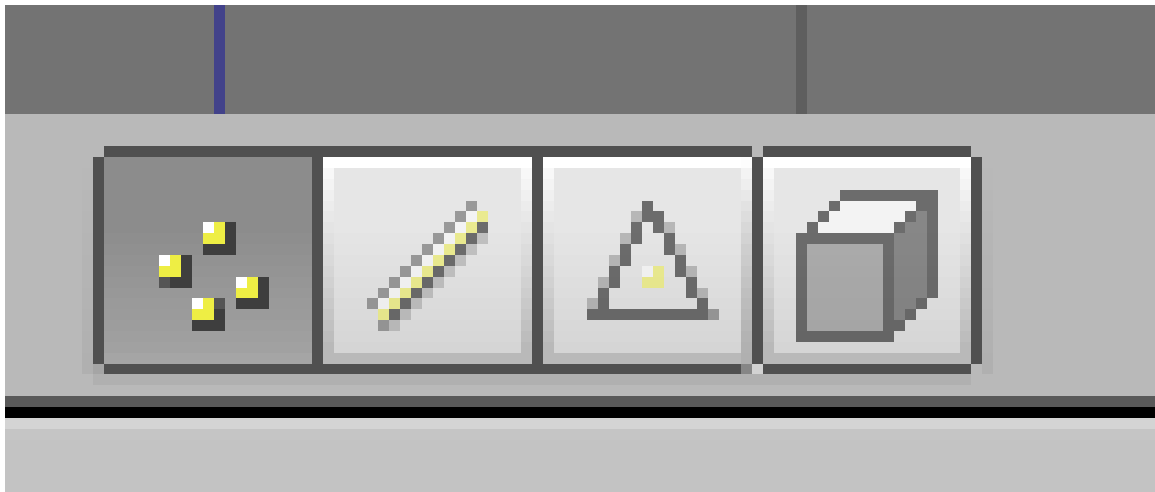


Figure 1321

- If you still selected just two vertices, change to a wireframe drawing by pressing **ZKEY**, which toggles between wireframe and solid drawing types. Try selecting again.
- Yet another way to select the six vertices is to select the two faces they define. Click the Face Select button from the selection mode buttons (above), or press **CTRL+TAB** to display a Select Mode menu and choose Faces. Rotate the cube to view the left side and click **RMB** in the center of one of the two upper faces. Hold **SHIFT** and click **RMB** in the center of the other upper face.

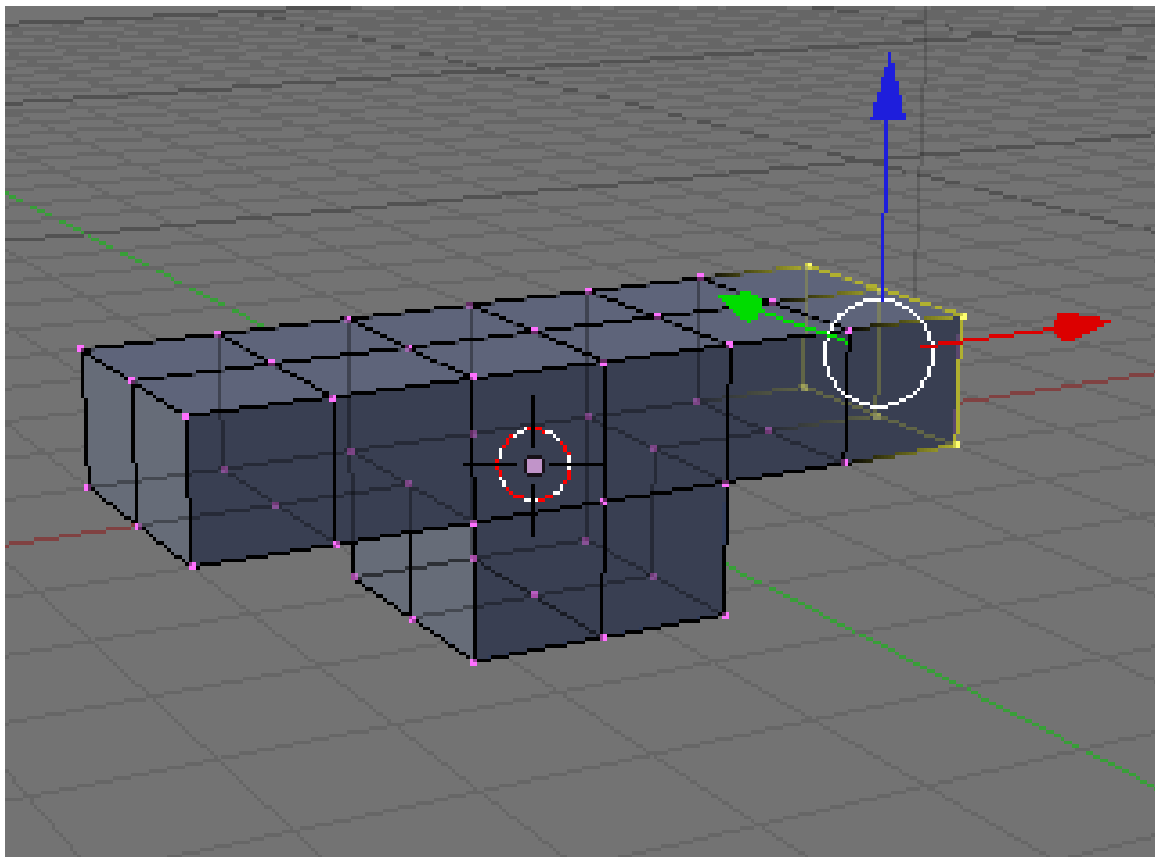


Figure 1322

202.1.3. Extruding Arms

- View from the front by pressing **NUM1**.
- Extrude the selected vertices/faces by pressing **EKEY** to display an Extrude menu, then choosing Region. Hold **CTRL** to snap your position to the grid, then move your mouse left to put the new vertices on the adjacent gray line of the grid one unit to the left. Click **LMB**.
- Repeat so that your model looks like below (from front view, **NUM1**).

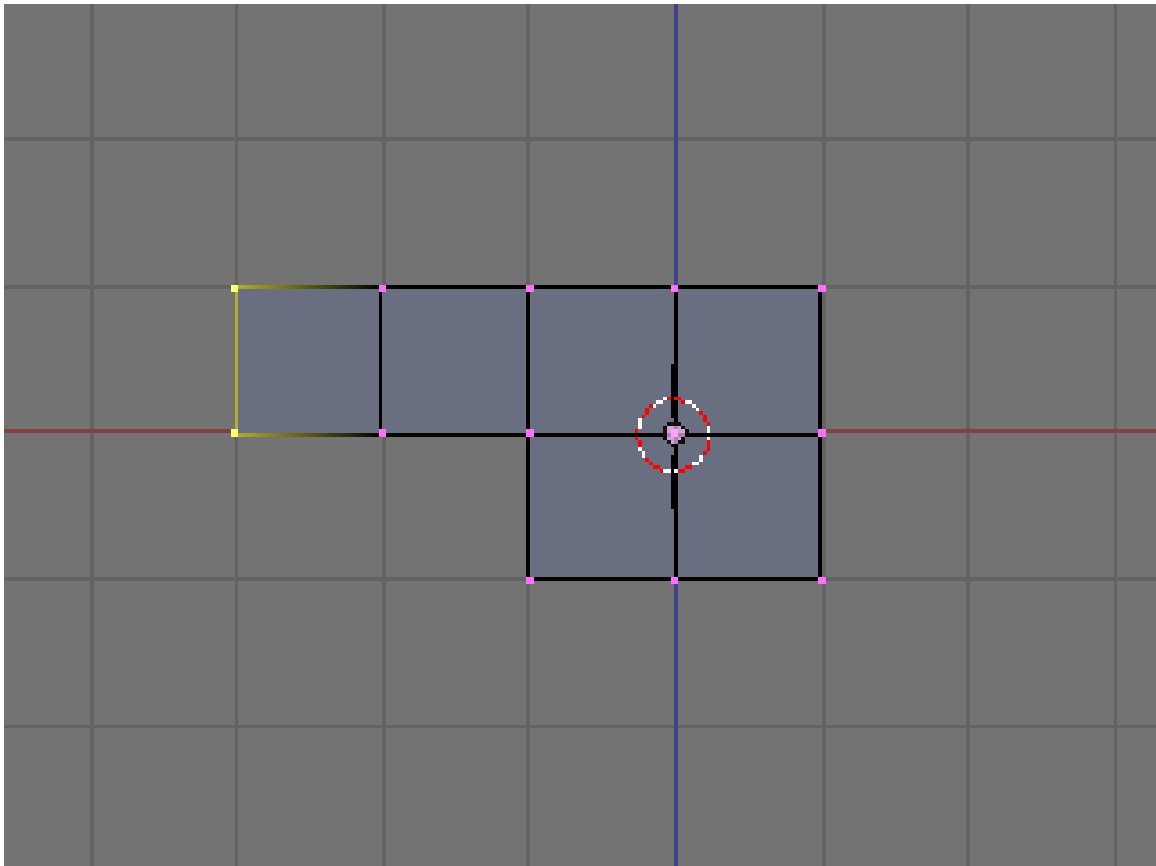


Figure 1323

- Deselect all vertices/faces with **AKEY**.
- Perform the same extrusion on the right side of the cube, selecting six vertices and extruding twice as explained above.
- The gingerbread man's arms are in place, as in the illustrations below.

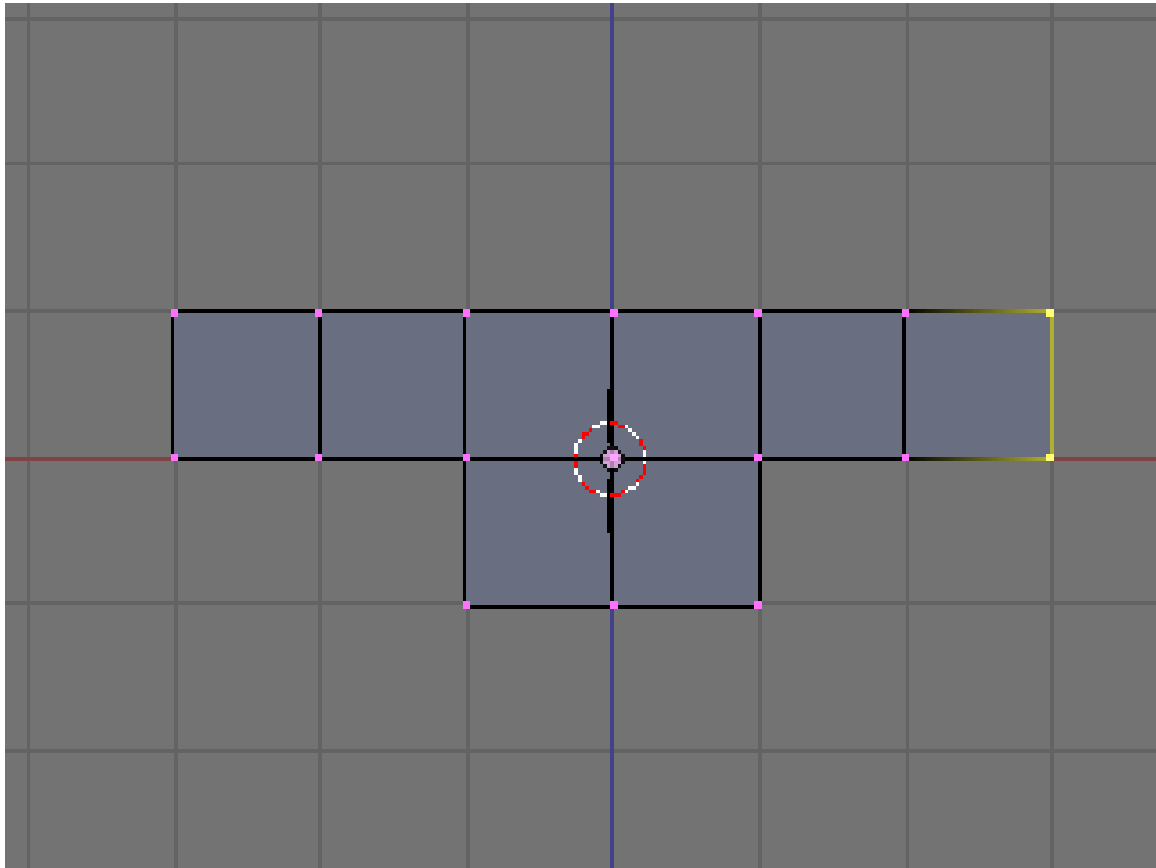


Figure 1324

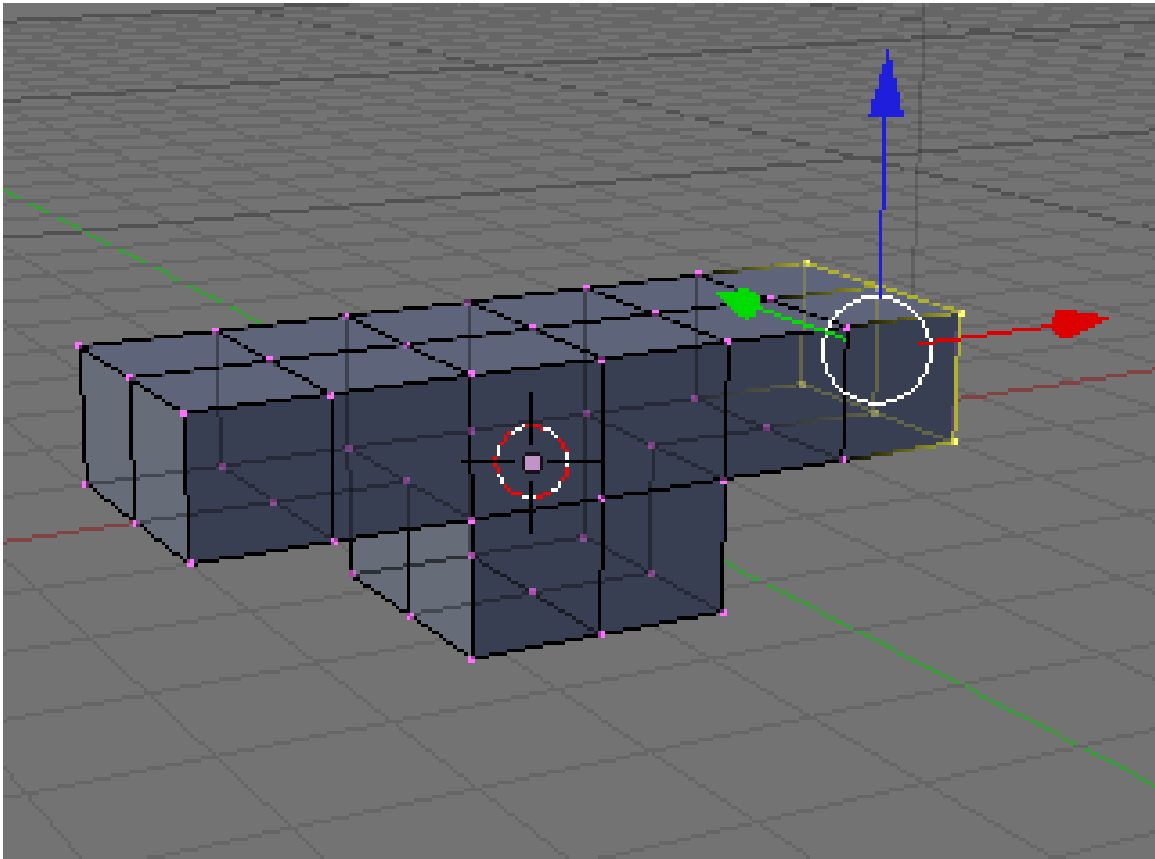


Figure 1325

202.1.4. Extruding Legs

- Make sure all vertices are unselected, using **AKEY** once or twice.
- View from the front (**NUM1**).
- Select the six vertices on the left half of the cube's bottom. Extrude them downward to a point in between the first and second heavier gray lines beneath the cube. (The gray lines in the grid represent Blender units.) Holding down **CTRL**, the extruded region may snap to the heavier gray lines—hold down **CTRL+SHIFT** and the region will snap to tenths of Blender units. Click **LMB** to finish.
- An alternative to positioning the extrusion with the mouse is simply typing the distance. Enter 1.5 to extrude 1 1/2 units out. On a Mac, enter the number 1, press fn with the key that is right under **LKEY** and **MKEY** on Azerty (the one with /:,), and press the number 5.

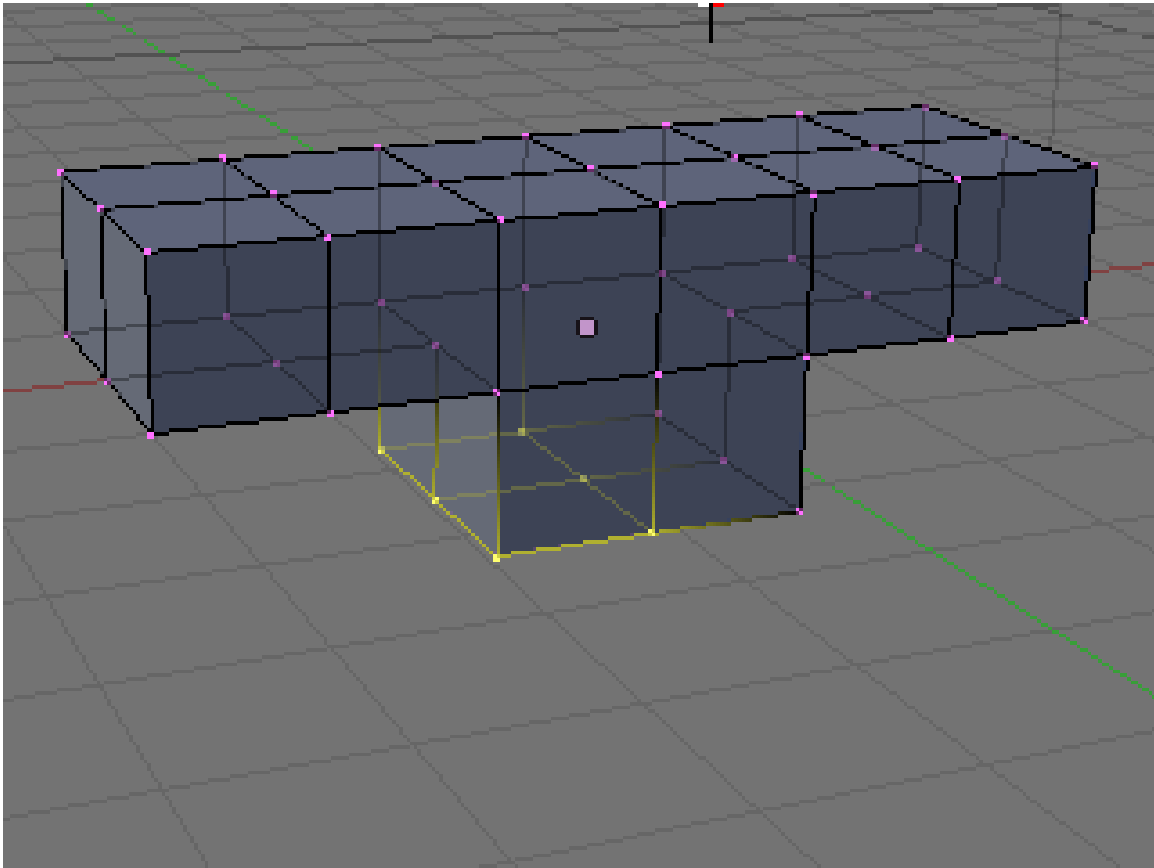


Figure 1326

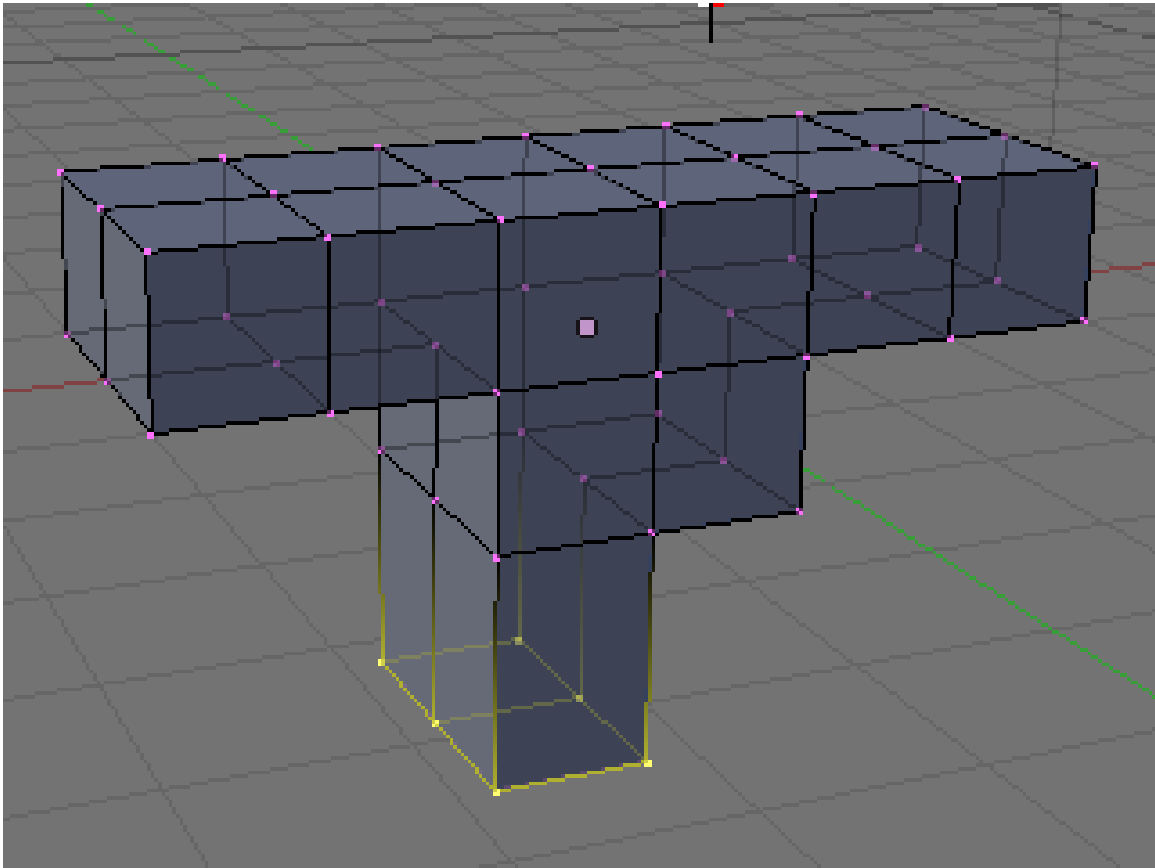


Figure 1327

- Extrude the same region again, to the third gray line (1.5 again). It should look like this:

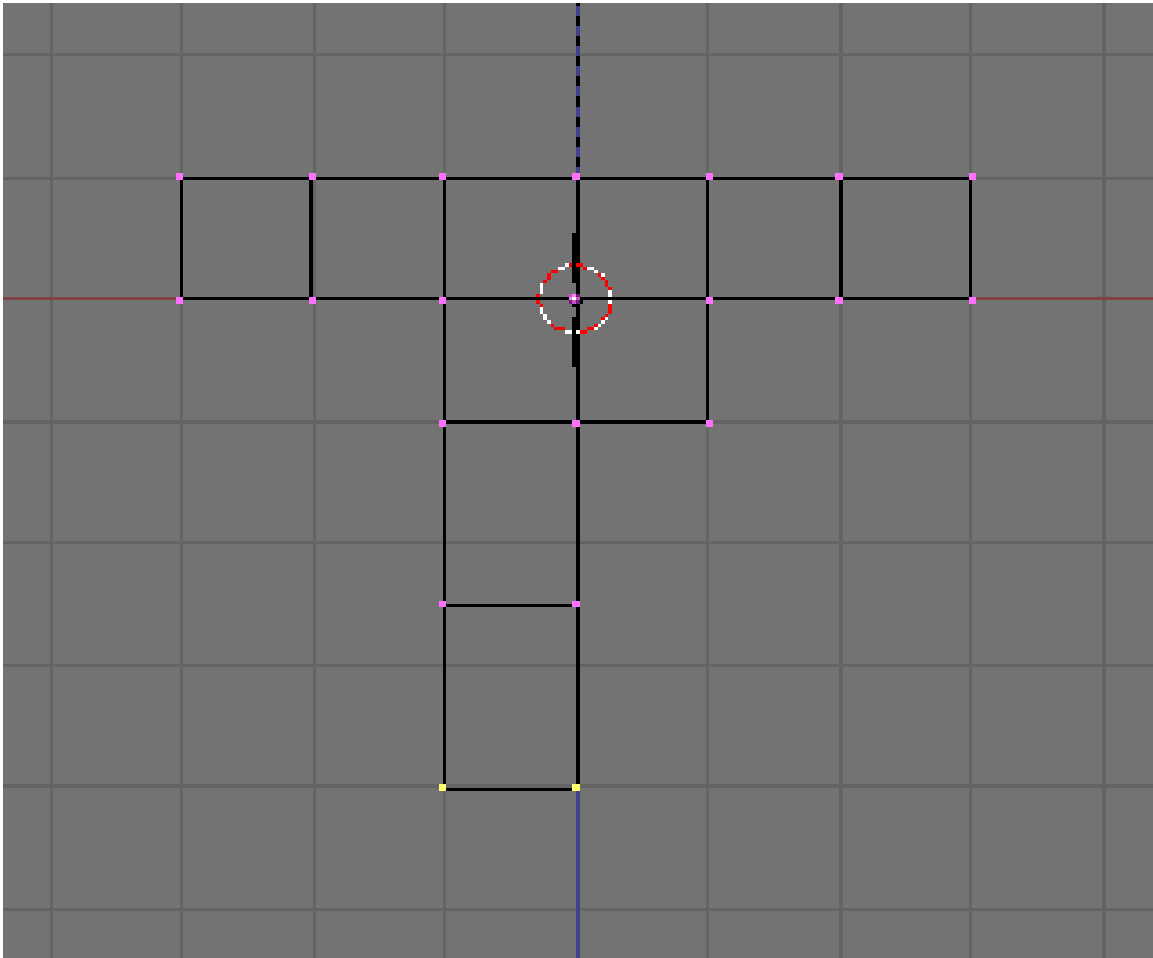


Figure 1328

- Shift the lower leg to the outside as follows.
 - Select the bottom 12 vertices of the leg (which look like 4 from front view), using **BKEY**.
 - Grab the selection with **GKEY**. Press **XKEY** to limit movement to the x axis. Move the vertices to the left by half a square, holding **CTRL** or **CTRL+SHIFT** to snap to the grid, and click **LMB**.

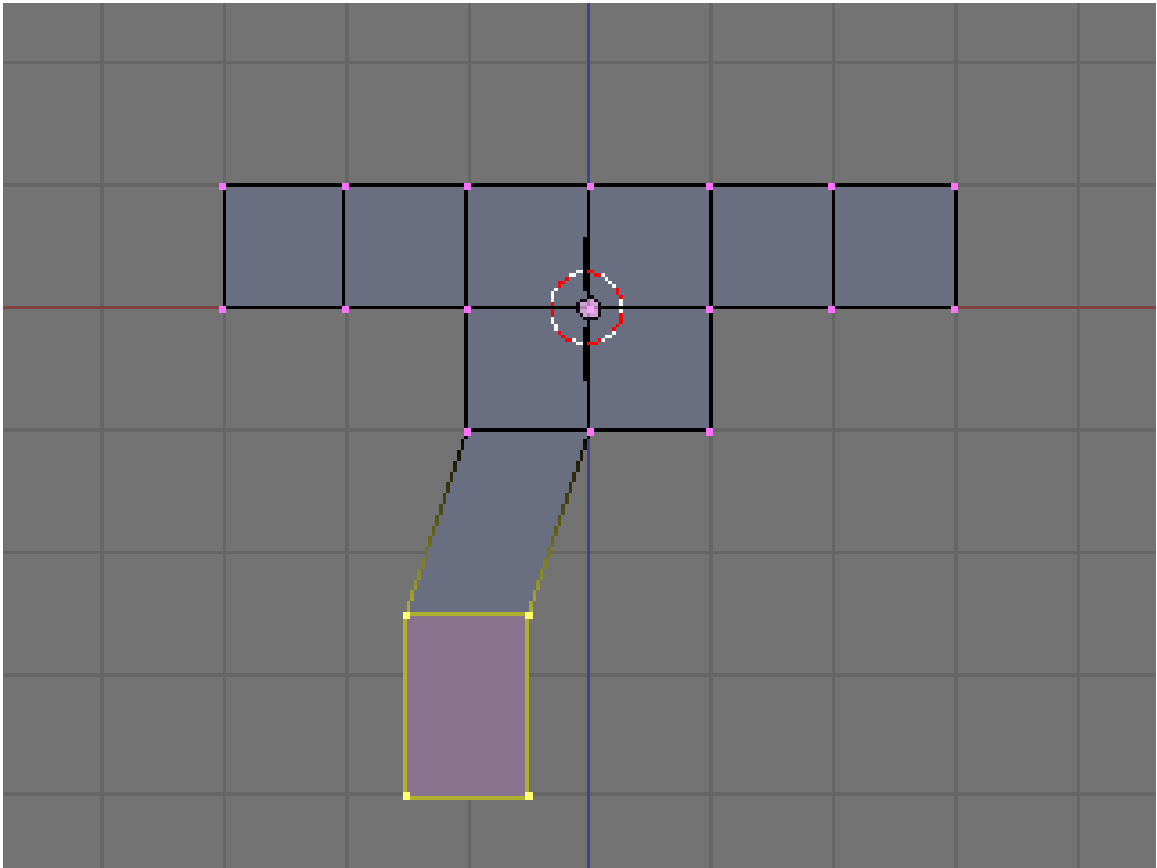


Figure 1329

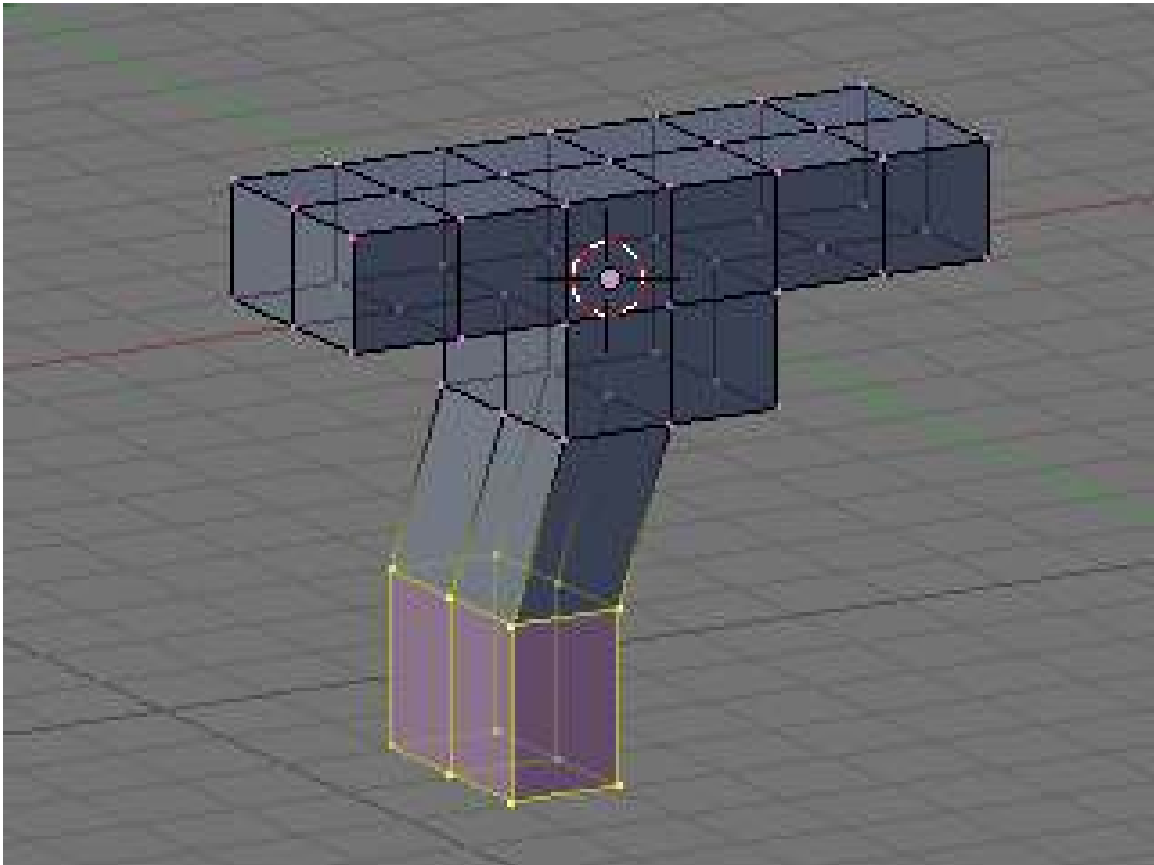


Figure 1330

- Create a second leg on the right side, in the same fashion.

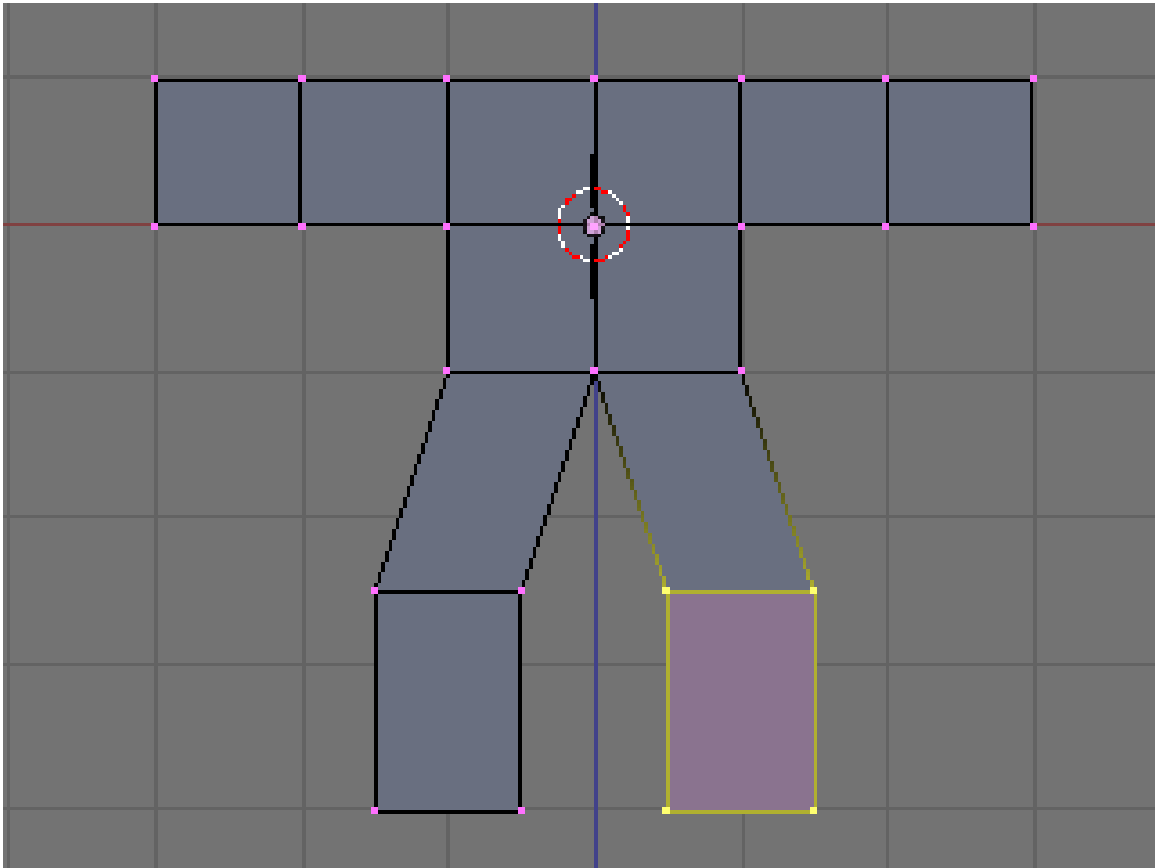


Figure 1331

202.1.5. Dropping the Groin

Lengthen the groin (where the two legs join):

- Deselect all vertices by pressing **AKEY**.
- Select the 3 vertices at the groin (which look like 1 from front view), using **BKEY**.
- Grab the vertices by pressing **GKEY**. Press **ZKEY** to limit movement to the z axis. Move the vertices down 1/2 of a square, or type -0.5 to specify the distance. (Older versions of Blender require **NKEY** before typing -0.5.)

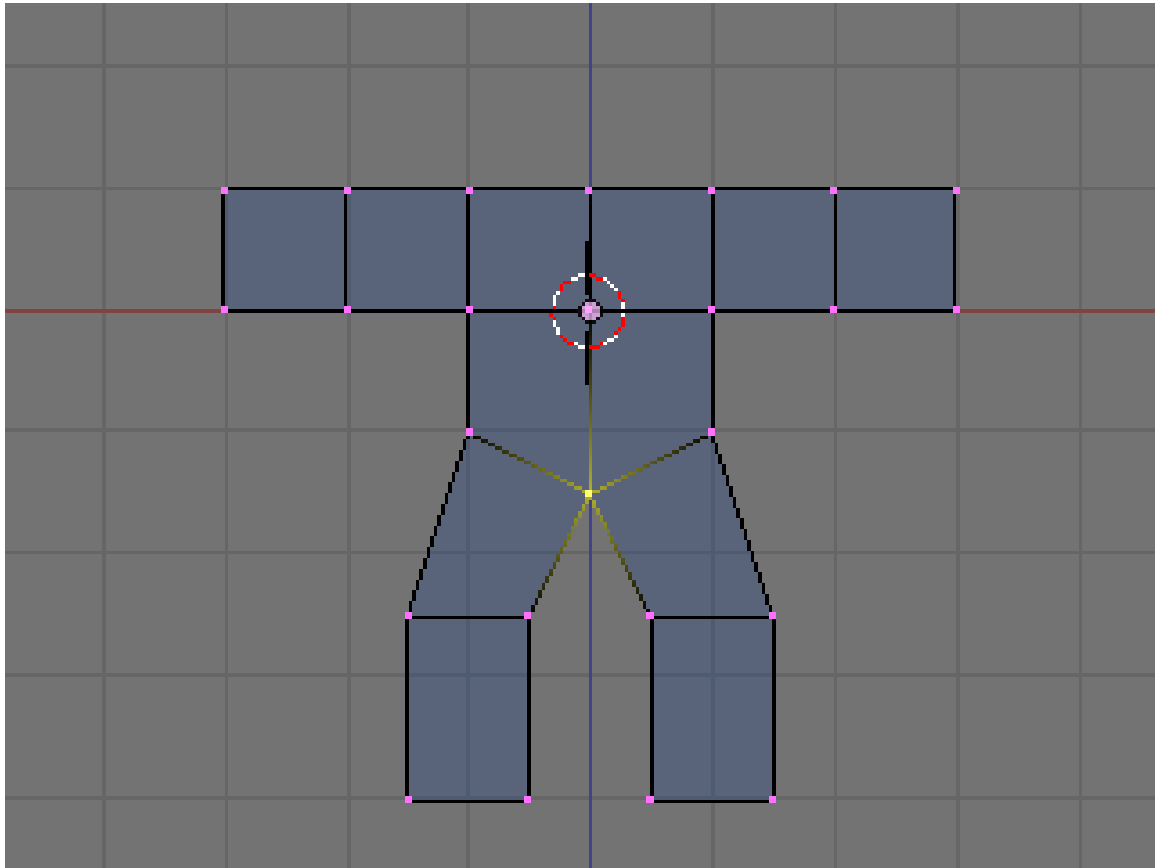


Figure 1332

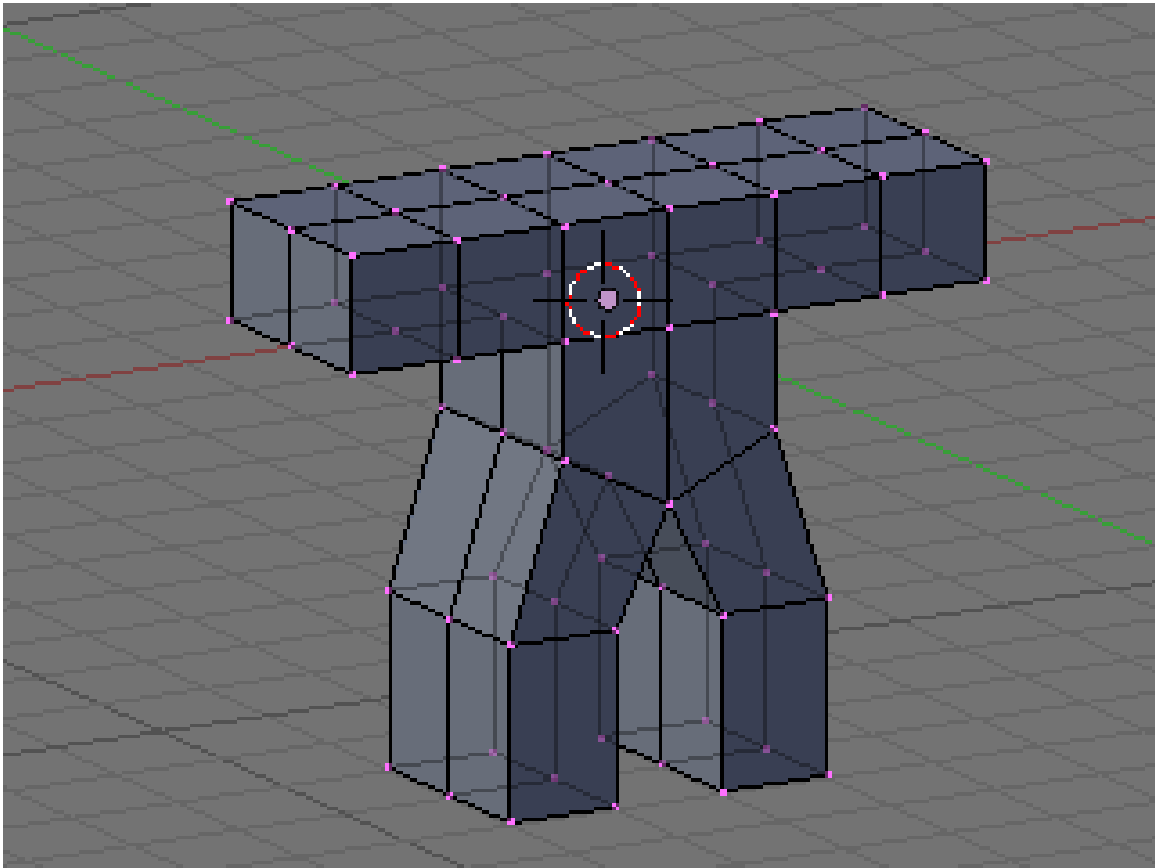


Figure 1333

202.1.6. Adding a Head

- Switch the 3D window from Edit mode back to Object mode by pressing **TAB**.

Noob Note: I found it easier to remain in edit mode while creating the cube as this makes the cube part of the original mesh. This allows you to add the subsurf to all parts at once.

- Click **RMB** on the object to select it then press **SHIFT + SKEY** and select *Cursor → Selection*. This will make sure the cube you'll add next will be near where you want it.
- Press **SPACE** and put your mouse on the mesh option and select cube. In others versions, you can also hit **SPACE** and , in the menu that comes up, choose *Add → Mesh → Cube*.
- Press **GKEY** and put your new cube about 1/3 of the way down the neck (to achieve this, you can press **GKEY** and **ZKEY** : enter 1.33).

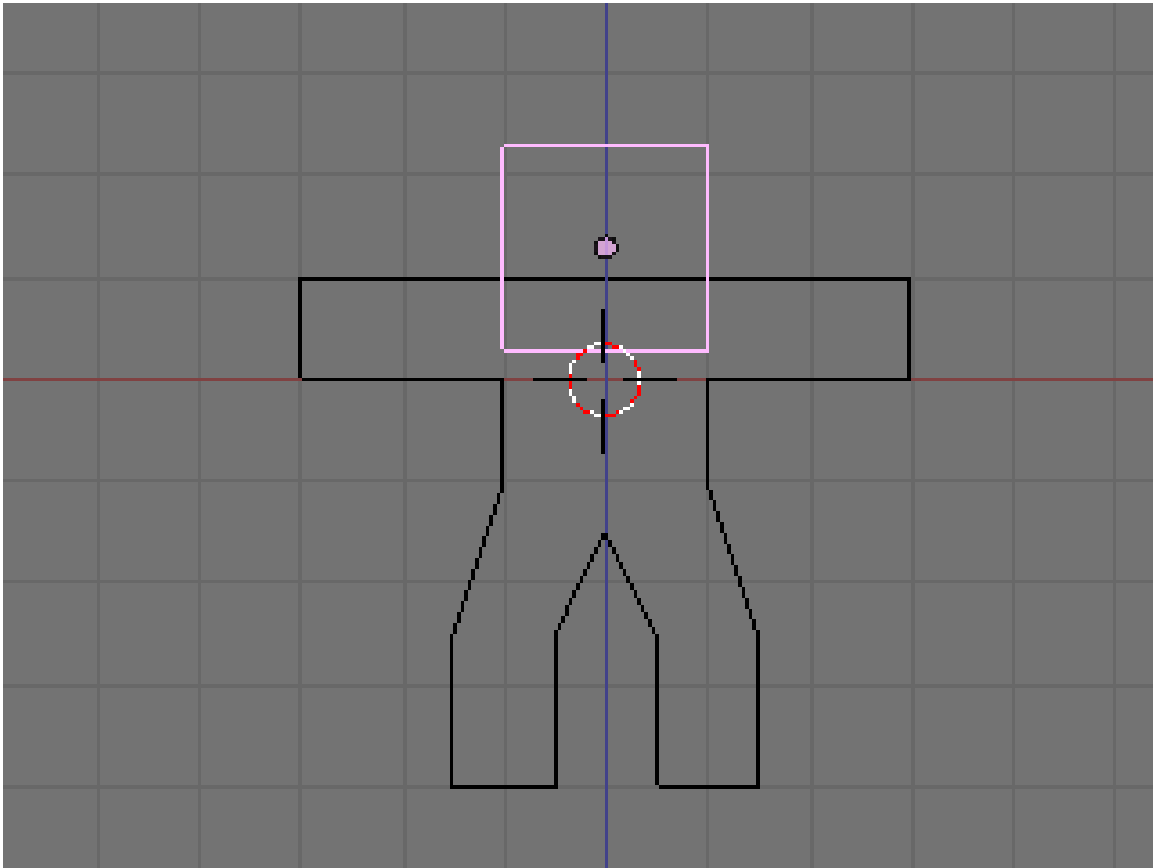


Figure 1334

Now we will make it look more like a ginger bread man by making it thinner.

- Select all with **AKEY**.
- Go to side view with **NUM3**.
- Press **SKEY** for scale and press **YKEY** for Y-axis and then move your mouse to the middle until it is about 0.3 (use **CTRL** for fixed values).
- Remember X-axis is the Red arrow/line, Y-axis is the Green one, and Z-axis is Blue (like RGB video mode).

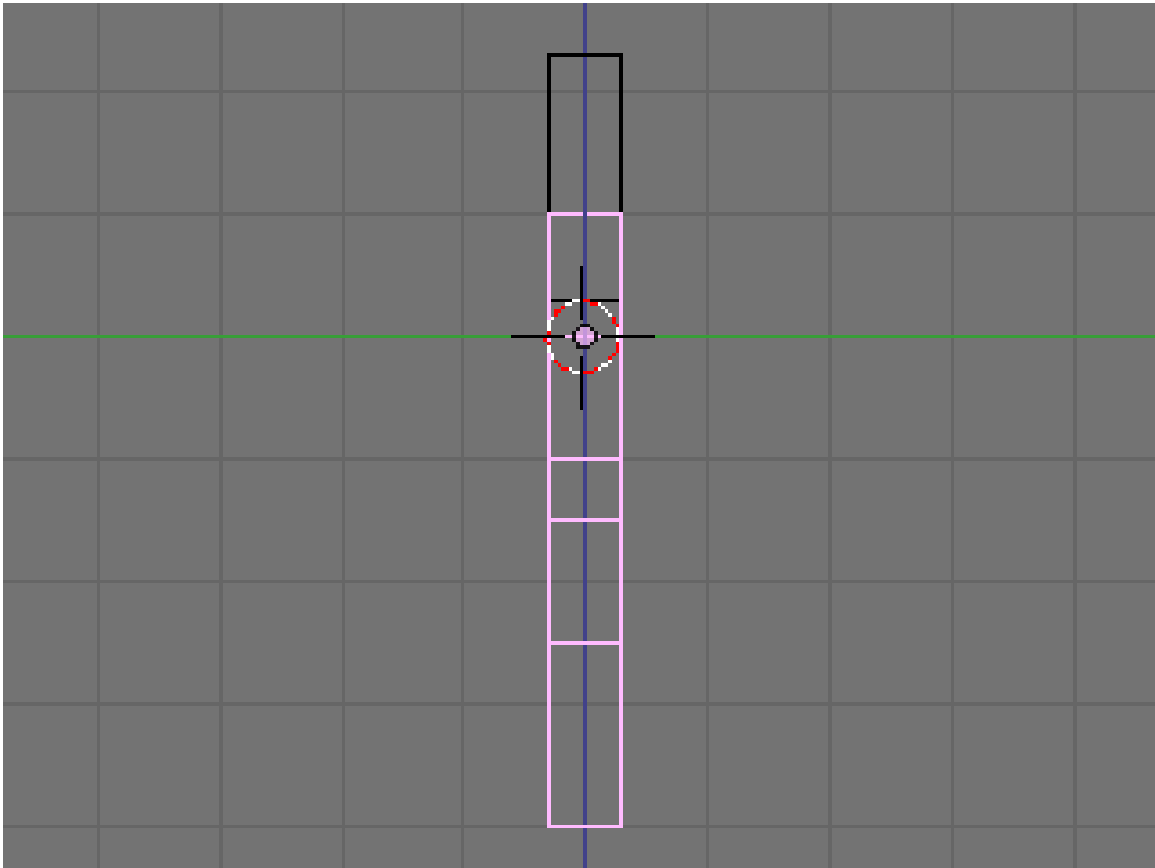


Figure 1335

- Use the **MMB** to spin the view around and examine your handiwork.

At this point, it doesn't look entirely like a gingerbread man, does it? It's a bit too ... chunky. For the last bit, we'll smooth it out.

- Make sure you've selected the body in object mode.
- Select the editing panel in the buttons window (or hit **F9**).
- In the Modifiers tab, Add a "Subsurf" modifier.
- Set the level of the subdivisions to 2, and the number of render levels to 3.

Noob question: When I add the cube for the head, it stops me from being able to edit the body - it will only select the head to apply subsurf to, even if the body looks like it's selected!

Answer: When you created the cube you made a second object. To select a different object, press tab to enter Object mode. Select the body. Then enter edit mode again if you want to edit the body.

- You can press the **ZKEY** to switch back and forth between wire-frame view and solid view.
- (Noob Note: Easiest way to really get a feel for what is going on in the 3d world is to split into four screens and setting each one to **NUM7**, **NUM3**, **NUM1**, and **NUM0** to see all angles and what it will look like at render.)

Noob Question: *How?*

Answer: To split an area move the cursor to an area between two current areas (e.g. between the 3D view and the buttons), when you see the double ended arrow (used to move the divide) click **RMB** and select *Split Area*, you will then see a line appear dividing the area in two. Move this to where you want the divide and click **LMB**.

- In the 'Link and Materials' section, select 'Set Smooth'.

*(Note that here I had the same problem as before, with superposed vertices. Select all vertices, then press **WKEY** and select Remove Doublesto clean your model. You will see that it will look much better after removing the extra vertices with Remove Doubles)*

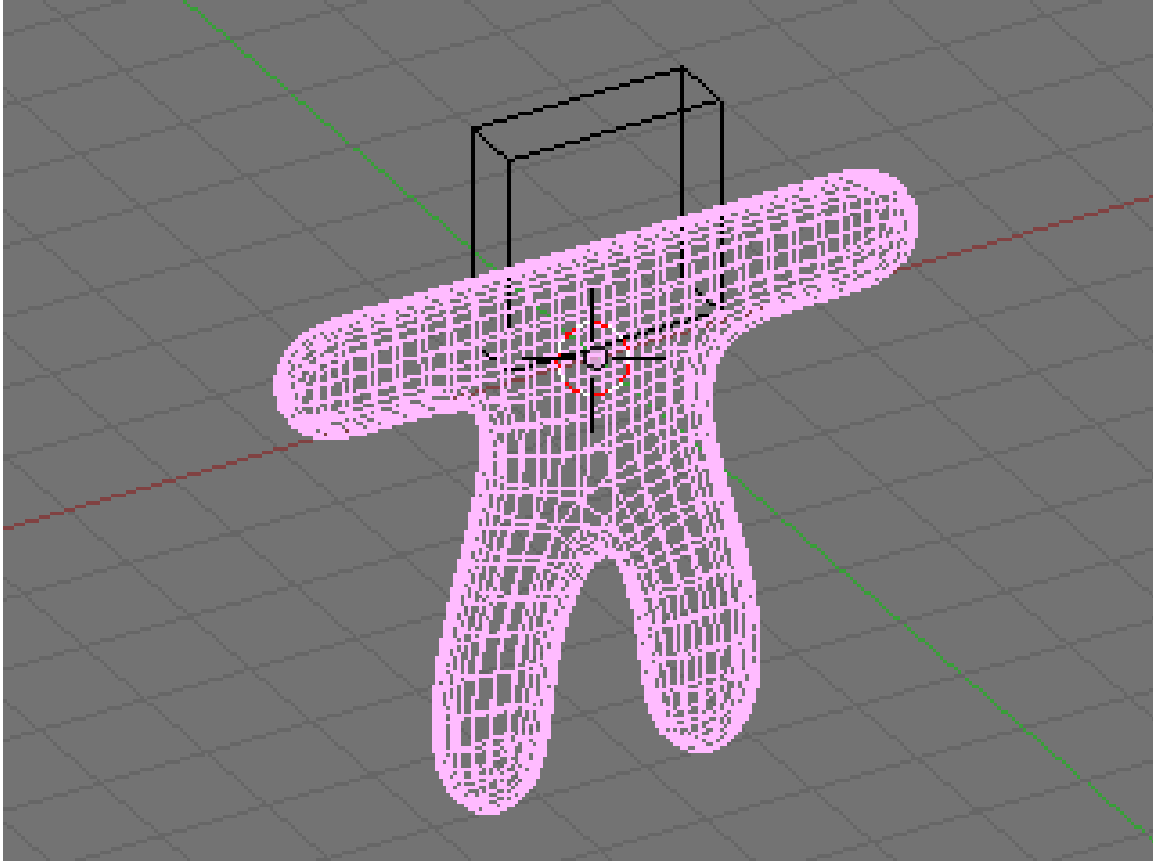


Figure 1336

- Press the **ZKEY** to return to wire-frame view.
- Now repeat the process above to smooth the head.

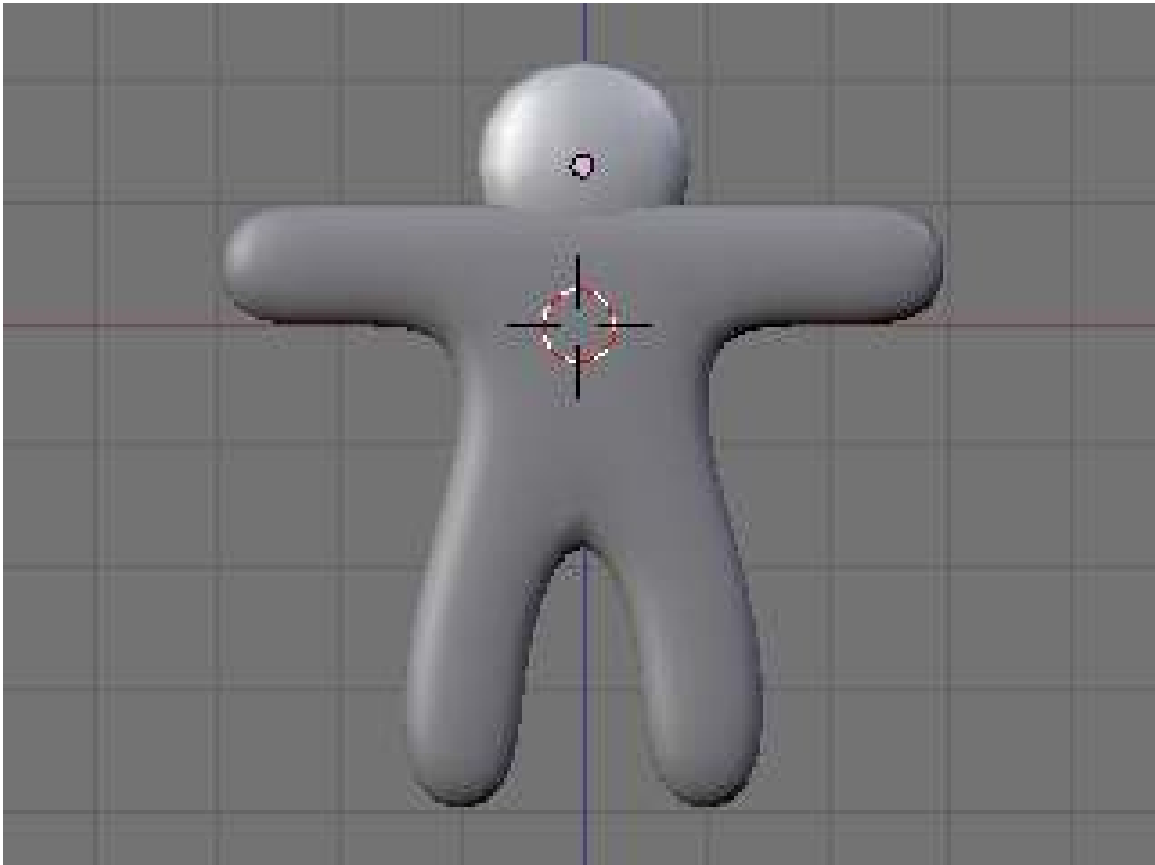


Figure 1337

Looks a lot more like a gingerbread man now, doesn't it?

202.2. Camera Positioning and Rendering

This guide will show you how to intuitively get the best frame of your 3D scene with no effort!

- Press **TAB** for Object view mode.
- Press **NUM0** to get the Camera View.
- Select the camera by clicking **RMB** on the outermost rectangle.
- Press **GKEY** and move your mouse to adjust the position of the camera (**XKEY**, **YKEY**, **ZKEY** and **CTRL** may be useful here).
- In addition, you can press **NUM7** to get the Top View and press **RKEY** to rotate the camera to the best angle.
- After you are happy with the position, press **F12** to render it.

If your render comes out a little dark, try moving the lamp closer to the gingerbread man.

Noob note: Another way to move around the camera is pressing **SHIFT + FKEY** after pressing **NUM0** to enter Fly mode. The keys for fly mode appear in the header of the 3D view pane.

Noob note: **Ctrl+Alt+NUM0** "teleports" the camera to your 3d view.



Figure 1338

Noob Note: By pressing X, Y or Z twice you will use a local base of the space, with those it's much easier. For example if you are facing the Z axis from 45 degrees, and you want to go left 1 unit, using the global base, you will have to go 1.72 (around $\sqrt{2}$) along X and the same along Y, instead moving by 1 in the local frame of reference.

202.3. Applying Textures

This builds on the previous guide: [Modeling a Volcano](#)¹.

Note: It seems that textures can only be applied to one object at a time, so this must be done twice (i.e. The head and body are two separate objects.) The settings that were chosen can successfully be applied to each object for a consistent result. Some settings can not be applied equally for consistent results.

N00b Note: if you join the 2 objects before applying the textures you do not have to do the process 2 times, just select the 2 objects, CTRL+J and the accept

¹ http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Modeling_a_volcano

- In "Object Mode," select the body (or the head.)
- Press **F5** to open the shading panel or use the shading panel button.
- In the "Links and Pipeline" panel, under "Link to Object," click "Add New."

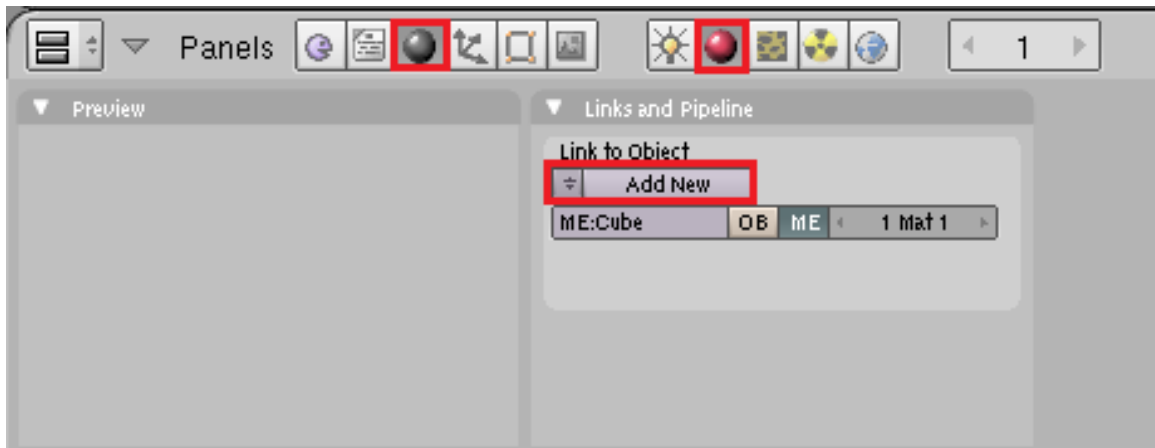


Figure 1339

- Press **F6** to open the "Texture Buttons" panel or use the textures button.
- In the "Texture" panel, click "Add New."

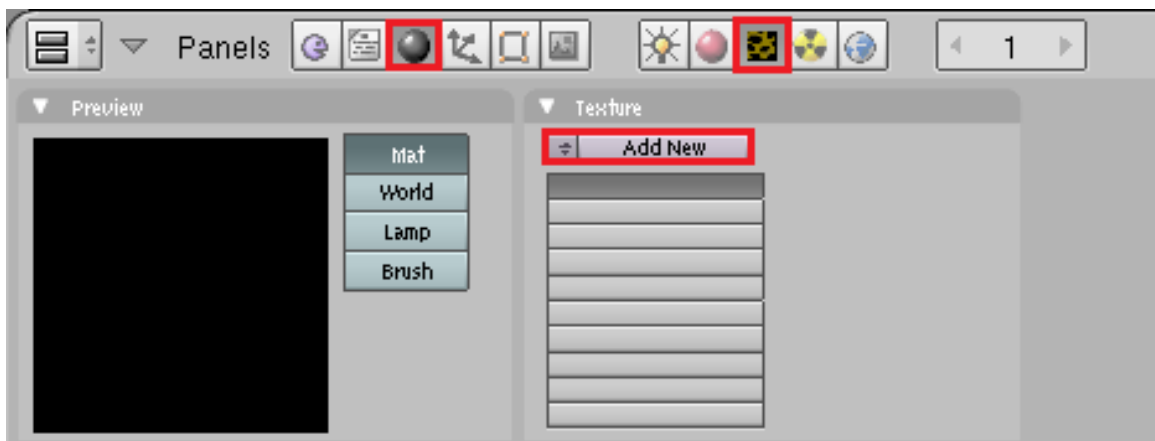


Figure 1340

- Change the "Texture Type" to "Stucci."

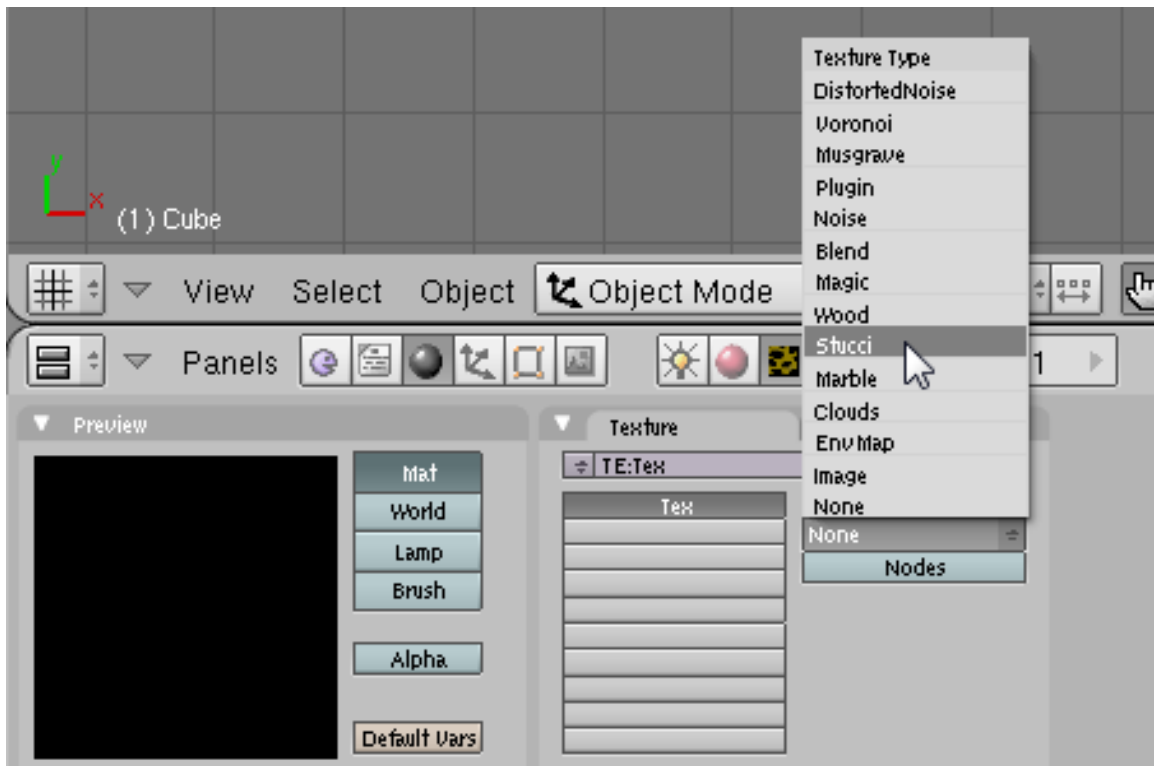


Figure 1341

- In the new "Stucci" panel, change "Noise Size" to something near 0.025 and leave the "Turbulence" at 5.00.

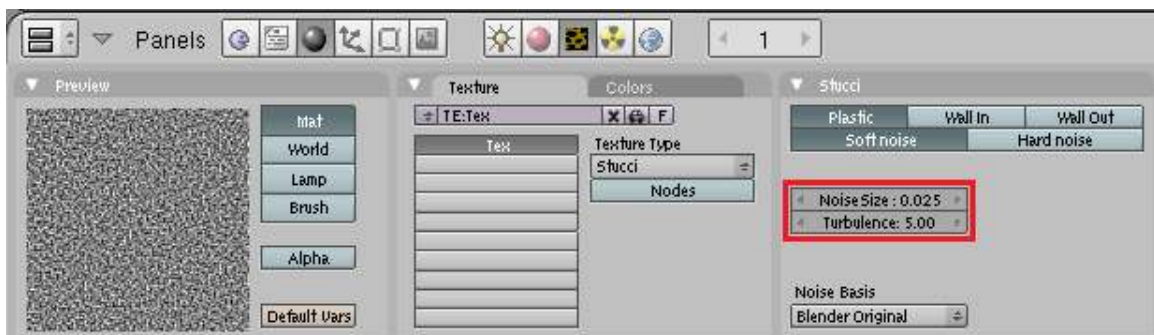


Figure 1342

Note: When finished with this section of the guide, come back to this panel and try different combinations of "Plastic," "Wall In," "Wall Out," and "Soft Noise" / "Hard Noise." Press **F12** to render after each change to see the effect.

- Press **F5** again or use the "Material" button directly on the left of the "Texture" button. Then look for the "Map To" panel.



Figure 1343

- In the "Map To" panel, deselect "Col[or]" and select "Nor[mal]."
- In the "Map Input" panel, change the texture coordinates to "Object" by clicking the corresponding button.
- In the "Material" panel, change the "R[ed]" slider to approximately 0.400 and the "G[reen]" slider to approximately half that, about 0.200. Blue can be set at 0.00.

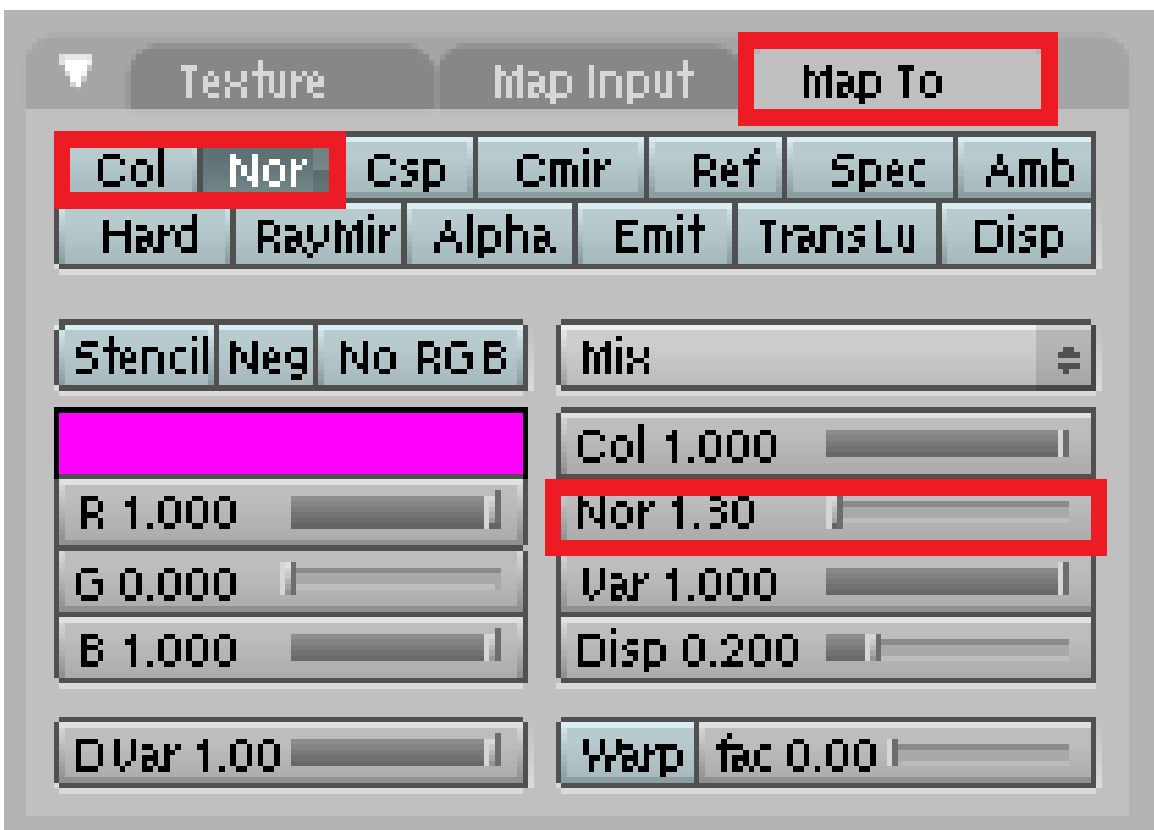


Figure 1344

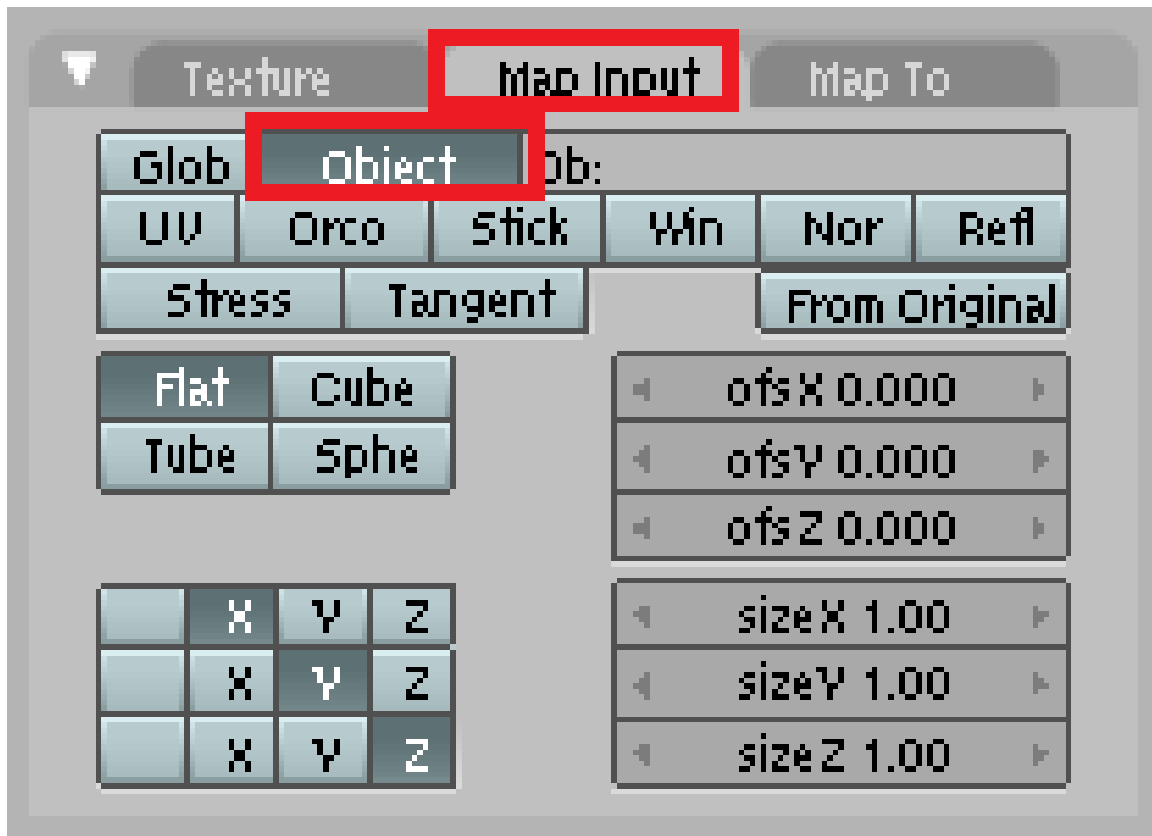


Figure 1345



Figure 1346

Note: While it is true that textures can only be applied to one object at a time, textures as well as materials can be shared between objects. In this case it is best to let both the head and the body share the same material.

- To do this, simply select the object without the materials(head or body).
- Press **F5** to open the shading panel or use the shading panel button.

- In the "Links and Pipeline" panel, under "Link to Object," click the arrow next to(left of) the "Add New" button.
- Select the brown material.

The steps in this section give a nicely textured, brown surface to the "Gingerbread."



Figure 1347

Now all you need to do is add eyes and gumdrop buttons!

203. Modeling a simple space-ship

203.1. Modeling a simple spaceship using Blender 2.49b

This is a simple tutorial on how to build a simple spaceship using blender 2.49b. One of the best things about blender is the adaptability that can be accessed through its user interface. This tutorial will give you a look at some of the included blender scripts. If you would like to make your own, however, you can refer to the scripting chapter of this wiki book. So now, I guess we can get started.

Step 1: First, open blender. You should get the default blender scene with the cube in the center. If you don't, try resetting your blender settings. Select the box in the center of the screen.

Step 2: Now we're going to shape the box into any shape we want. For now, we are going to shape it into a sort of triangle. If you want to use it as a box, that's fine, but I suggest following this tutorial using it as a sort-of triangle. To be able to modify it, in the lower buttons area (with stuff like Modifiers, Shapes, Multires, Mesh, and Link and Materials) you should see a drop-down menu that is currently on Object Mode. Change that to Edit Mode or hit the **tab** button. Now you will be able to modify different aspects of your box.

Step 3: With your box selected, hit Ctrl+Tab, and choose Faces. This allows you to modify the different faces. Faces are basically the different sides of your object, Edges are the edges of your object, and Vertices are points on your object where edges connect.

Step 4: Now, on your box, right click on one of the sides to select that side. If you want, first take a look around the box and then choose a side, but don't select the bottom or top face. To rotate around the box, hit the 8, 2, 4 or 6 button on your number pad, the one to the right of the arrow keys usually. Those buttons allow you to rotate around your objects and change your view. After you select the face you want, then rotate so that the face is facing towards you and looks like a square, and so that the grid around it looks like a flat horizontal line.

Step 5: There should be three arrows converging in the center of your face (or in the center of your box), and either the red or the green one should now be in the center of your face, and the other should be going off to the left or right, with the blue arrow going straight up or straight down. Use the 4 or 6 number pad button to rotate your view so you can see whichever arrow was in the center of your face, and left click mouse button on it, and hold the left click button, and drag the arrow around. Drag it until you have a rectangle. <http://i238.photobucket.com/albums/ff55/bryguy336/uh.png>
<http://i238.photobucket.com/albums/ff55/bryguy336/box.png>

Step 6: Now to resize that face. With that face still selected, hit the S key, and move the mouse around. This should enable you to change the size of the face, and once you click, it should stop resizing and stay at that size. Experiment and get the hang of it, and then resize it so that it is an itsy bitsy teeny tiny box, and your rectangle more like a triangle. Congratulations! You have just made your first shape/object

204. Creating Weapons based on 2D Images

Alright, this is my first tutorial here, I made this a while back, but hopefully it will be a good addition to this Wikibook. It was originally made for a game called *Fable - The Lost Chapters*¹, but I think it is worthy of another place in the world. Please note that this is a WIP and will be updated time to time.

Screenshots coming soon.

*You can find a high quality video here.*²

204.1. Modeling Technique 1

Modeling Technique 1 Part 1 Video³

Part 1:

1. After emptying the screen, go to view and click background image, load, then choose a picture of the weapon you want
2. Go to top view and add a square, then delete two of the vertices and place one of the remaining on a point on the picture, and the other out of the way.
3. Select both vertices and subdivide.
4. Move the generated point to the next good-looking spot on the picture
5. With that selected still, select the outside vertex also, and subdivide.

(Note: A much easier way is to select one vertex, move it to where you want it, select the other vertex, move it to the next spot, then extrude the second vertex to the next point, rinse and repeat. This requires a lot less effort.) Another guy's note: For an easier way, just click CTRL+LMB where you want the next point to appear.

1. Continue until you are on your third to last one, then select your outside vertex and move it where you would move the next vertex.
2. Select the first and last vertices and go to mesh>>make edge/face
3. Extrude to half your preferred thickness, then extrude the rest of the way.
4. Select the center vertices on the blade, and scale up.

Modeling Technique 1 Part 2⁴

Part 2:

-
- 1 [http://en.wikipedia.org/wiki/Fable_\(video_game\)](http://en.wikipedia.org/wiki/Fable_(video_game))
 - 2 <http://files.filefront.com/model+technique1+higher+qyavi/;9063237;/fileinfo.html>
 - 3 <http://youtube.com/watch?v=isQzBvPGIx8>
 - 4 <http://youtube.com/watch?v=R3UgOXYZMX4>

cont.)

1. Then move vertices to your liking.
2. Select the vertices of the handle, go to mesh>>vertices>>separate.
3. Add a modifier, choose subSurf, turn up the level until it looks close to what you want without too many vertices. Click apply next to the modifier.
4. Modify vertices (using proportional edit helps) to your liking.
5. Go to object mode, and turn off double-sided on any meshes that are.
6. If black appears on any parts of the mesh, highlight it, go to edit mode, select all vertices, go to mesh>>normals>>recalculate outside.
7. If there is still black, select those faces and go to mesh>>normals>>flip.
8. If there is still black, then you are missing a piece of mesh. Highlight the vertices around the hole, go to mesh>>vertices>>fill.
9. If black did not appear, then select everything (in object mode) and go to object>>join objects and say yes.

204.2. Modeling Technique 2

Video [Here](#)⁵ Download it for high quality.

Blade:

1. Select all with the 'A' key, and delete everything.
2. Go to view and choose background image >> use background image >>load and pick the sword you'll be doing.
3. Close the dialog after you see it in the background (you can change the brightness of it by changing the 'blend' option)
4. Make a bezier circle by pressing spacebar and going to add>>bezier>>bezier circle.
5. put a bar at each main point of your weapon (anywhere the curve changes direction)

Bezier Controls⁶

1. Modify the bars until they match.
2. You might want to go into wireframe view.
3. Go down and turn up the bevel depth a little, to give it some sharpness.
4. To lower the poly count you should turn down the DefResolIU number.
5. Go back to object mode, hit spacebar, go to object>>convert object type>>to mesh.
6. Modify vertices to your liking.

For the handle:

1. Do steps 1-7.
2. Turn up the bevel resolution, to give it some roundness.
3. Repeat steps 8-10

For That one thing (see video):

5 <http://files.filefront.com/modeling+tecnique+2+agkavi/;9221627;/fileinfo.html>

6 <http://wiki.blender.org/index.php/Manual/Curves#B.C3.A9ziens>

Just watch the video, but here is a summary.

Extrude and flatten a circle, use proportional edit to make it easier on you when you curve the circle.

For hilt guard:

1. Repeat steps 1-6
2. Add some extrusion.
3. Repeat steps 8-10.

Morerunes⁷ (talk⁸) 17:52, 18 December 2007 (UTC)

⁷ <http://en.wikibooks.org/wiki/User%3AMorerunes>

⁸ <http://en.wikibooks.org/wiki/User%20talk%3AMorerunes>

205. Match Moving

Match moving is the technique of recreating the position of the camera used in recording live action footage. This information can then be used within Blender to merge 3D objects with live action film. For a more detailed discussion of the concept, take a look at the Wikipedia article on Match Moving¹.

Blender cannot perform match moving itself, you must use a 3rd party tool to determine the camera position and the way it moves, then import this data into Blender. While there are many software tools to do this, this page references two free options: Voodoo and Icarus.

205.1. Icarus

Icarus² is a discontinued University of Manchester project which can be used for non-commercial work. The download links from the official page no longer function, but Windows and MacOS X versions are available from this Icarus video tutorial³ by Colin Levy.

205.2. Voodoo

Voodoo⁴ is an actively developed free match mover available for Windows and Linux. Here is a tutorial on using Voodoo and Blender⁵

1 <http://en.wikipedia.org/wiki/Match%20Moving>
2 <http://aig.cs.man.ac.uk/research/reveal/icarus/>
3 <http://peerlessproductions.com/tuts/pages/Icarus.html>
4 <http://www.digilab.uni-hannover.de/docs/manual.html>
5 <http://fictionality.co.uk/doku.php?id=tutorials:matchmoving>

206. Motion Tracking with Icarus

Motion tracking, also called **Match moving**¹, is an essential element when integrating 3D elements with live footage. Motion tracking software is usually pretty expensive, but the Icarus application (Windows and Mac) is available for free for educational use. Icarus, which hasn't been updated for while, was later replaced by the commercial application PFTrack². Other popular motion tracking applications are PFMATCHIT³ and PFHoe⁴ (both also from The Pixel Farm), Voodoo⁵ (for Windows/Linux; free for non-commercial and commercial use), SynthEyes⁶, Boujou⁷ and 3D-Equalizer⁸ (commercial).

The excellent CG prodigy Colin Levy⁹ hosts Icarus (by kind permission of The Pixel Farm Ltd), the Icarus import script for Blender, as well as a splendid video tutorial (see Download Icarus and Video Tutorial¹⁰). However, I lacked a brief text tutorial about motion tracking, so I decided to write my own. This tutorial is extremely brief and high-level, and requires some previous knowledge on video editing, 3D, and Blender.

Note: this tutorial was created using Mac OS X 10.5 Leopard, Blender 2.46, Icarus 2.09, and the Icarus Import Script for Blender¹¹ v1.07e (for Blender 2.41, written by Alfredo de Greef).

206.0.1. Tutorial

Phase 1: Preparing the Video Footage

Note: this tutorial explains the *Auto-feature Tracking* mode in Icarus. There are other options which gives more user control - see the Icarus **UserGuide.pdf** for more information.

1. Record your video footage. Having the camera on a tripod (thus limiting to just panning/rotating) simplifies the tracking, but Icarus can handle a hand-held camera as well. Filming a background with orthogonal lines (that can align to X/Y/Z dimensions), such as a room, also helps the tracking.
2. Capture/import your video footage to your computer. Icarus handles video up to DV resolution (720*576 pixels).

1 http://en.wikipedia.org/wiki/Match_moving
2 <http://www.thepixelfarm.co.uk/product.php?productId=PFTrack>
3 <http://www.thepixelfarm.co.uk/product.php?productId=PFMatchit>
4 <http://www.pfhoec.com/>
5 <http://www.digilab.uni-hannover.de/docs/manual.html>
6 <http://www.ssontech.com/>
7 <http://www.2d3.com/product/?v=1>
8 <http://www.sci-d-vis.com>
9 <http://www.peerlessproductions.com>
10 <http://www.peerlessproductions.com/tuts/pages/Icarus.html>
11 http://www.peerlessproductions.com/tuts/IcarusTutorials/ICARUS_import241.zip

3. Start the Icarus *Calibration* application (there is also a *Distortion* and a *Reconstruction* application).
4. Create a new project (Project->New).
5. Import your video footage (Project->Import Movie).
6. Fill in the *Camera Parameters* information in the window that pops up - especially the *Camera Motion* and the *Pixel Aspect* options.
7. In the left panel, expand the group called *Coordinate Frame*. You should see X Axis, Y Axis, etc.
8. Click the Z Axis tool (blue) and mark vertical lines in your video footage. Use the X Axis (red) and Y Axis (green) tools to mark horizontal lines (up to you to decide which should be X and Y).
9. Estimate the focal length (Camera->Estimate Focal Length).
10. Navigate in time in your video footage using the time slider (beneath the video image). Add more X/Y/Z marker lines on a few key frames, especially as new pieces of the background are revealed when the camera moves.
11. Save your project (Project->Save).
12. Start the tracking process (Camera->Track and Calibrate). This will take some time.
13. Export the results in human-readable form (Project->Export 3D Motion, select *Human Readable (*.txt)* as file type).

Phase 2: Importing the Motion Tracking Data into Blender

1. Start Blender, and open a Text Editor view.
2. Open the Icarus import script **ICARUS_import241.py** (File->Open).
3. Start the script (File->Run Python Script). You should now see the Icarus Import screen.
4. Press the *FSEL* button, and open the results you exported from Icarus.
5. Press the *Create Curves* button. This imports the camera motion from the Icarus data and applies it to the Blender default camera.
6. Press the *Feature Points Mesh* button. This imports 3D shape dots from the Icarus data, which helps as reference when you want to align your own 3D elements to the video footage.

You are now ready to add your own 3D elements to the Blender scene.

Phase 3: Compositing 3D Elements on top of Video Footage

If you want to easily composite the 3D elements on top of an image, you can add the image as the rendering back buffer in Blender (Scene tab in the Buttons view). However, this doesn't work for videos, so we need another solution.

1. In Blender, switch to *SR:4 - Sequence* in the layout dropdown menu at the top of the screen.
2. In the *Video Sequence Editor* view (middle of screen), add your video file (Add->Movie). Move the new strip to layer 1, frame 1.
3. Add the current scene to the sequence (Add->Scene, Scene). Move the new strip to layer 2, frame 1.
4. Select the scene strip on the second layer (right-click).
5. Open the *Scene* panel in the Buttons view, and then open the *Sequencer* sub-panel.

6. Change the *Blend Mode* dropdown from *Replace* to *Alpha Over*. Your 3D elements should now render over the background video in the top-right preview screen.
7. In the Render panel, enable *Do Sequence* just below the ANIM button. This will enable the background video when rendering.

206.0.2. Troubleshooting

- If your imported Feature Points Mesh looks a bit spherical, you need to generate camera distortion data using the Icarus *Distortion* application.

207. Create a Clayman

207.1. Create a Clayman

(under construction)

This Tutorial was made using Blender version 2.41

Upon completing this tutorial, you will have made something like this:



Figure 1348

Completed Clayman Render

207.2. Making The Clayman

Upon start, delete the default cube (X-KEY) and switch to front view with NUM1. Now add a mesh plane and under The Edit Buttons Window, Mesh Tools Tab click the subdivide button. Now select the 3 vertices on the left and delete them. (B-KEY for Border Select Tool)

If you can't select the vertices then deselect the Limited Selection to Visible button.

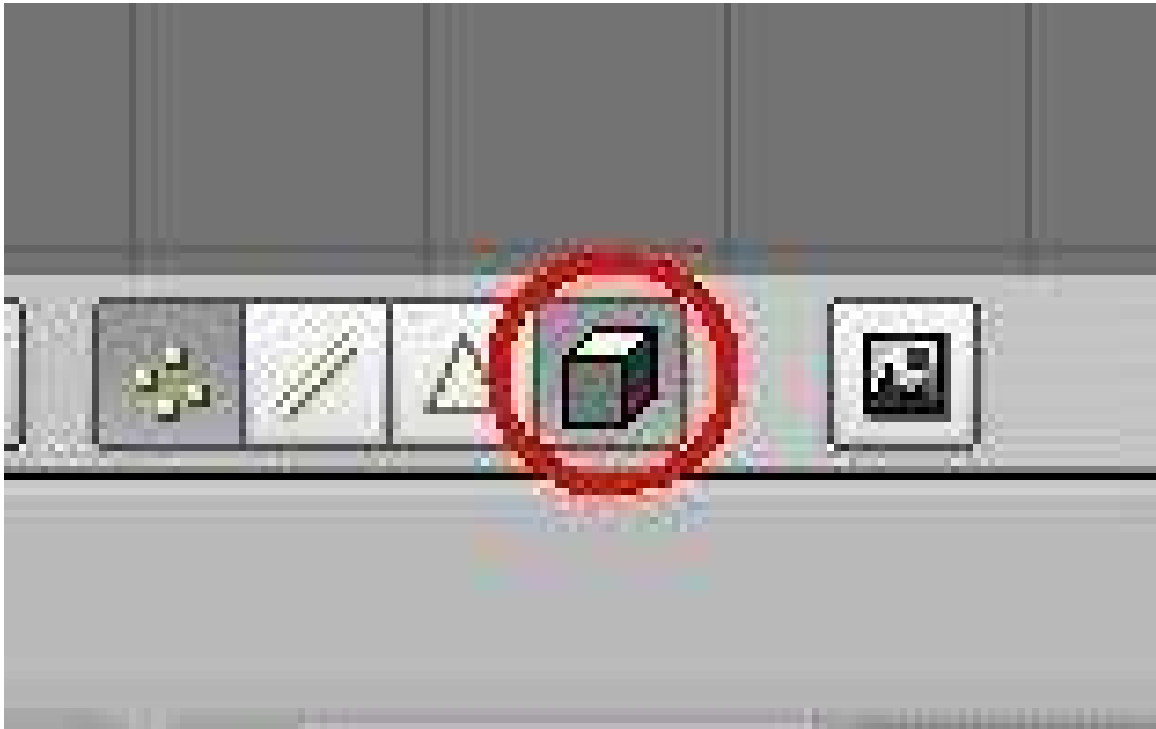


Figure 1349

Now select/deselect all vertices with A-KEY (we want them selected) In the Modifiers Tab in the Edit Buttons Window, Click *Add Modifier > Mirror* You should now see something like this.



Figure 1350

Now select the 3 right vertices and extrude them on the X axis by movement of 1.0 units by pressing E-KEY (*Extrude > Only Edges*), X-KEY (to constrain it to the X axis) and holding CTRL to constrain it to simple movements. Do this twice.



Figure 1351

Now extrude the legs by selecting the 3 bottom-right vertices and extruding them on the Z axis by 1 unit. Do this twice as well.

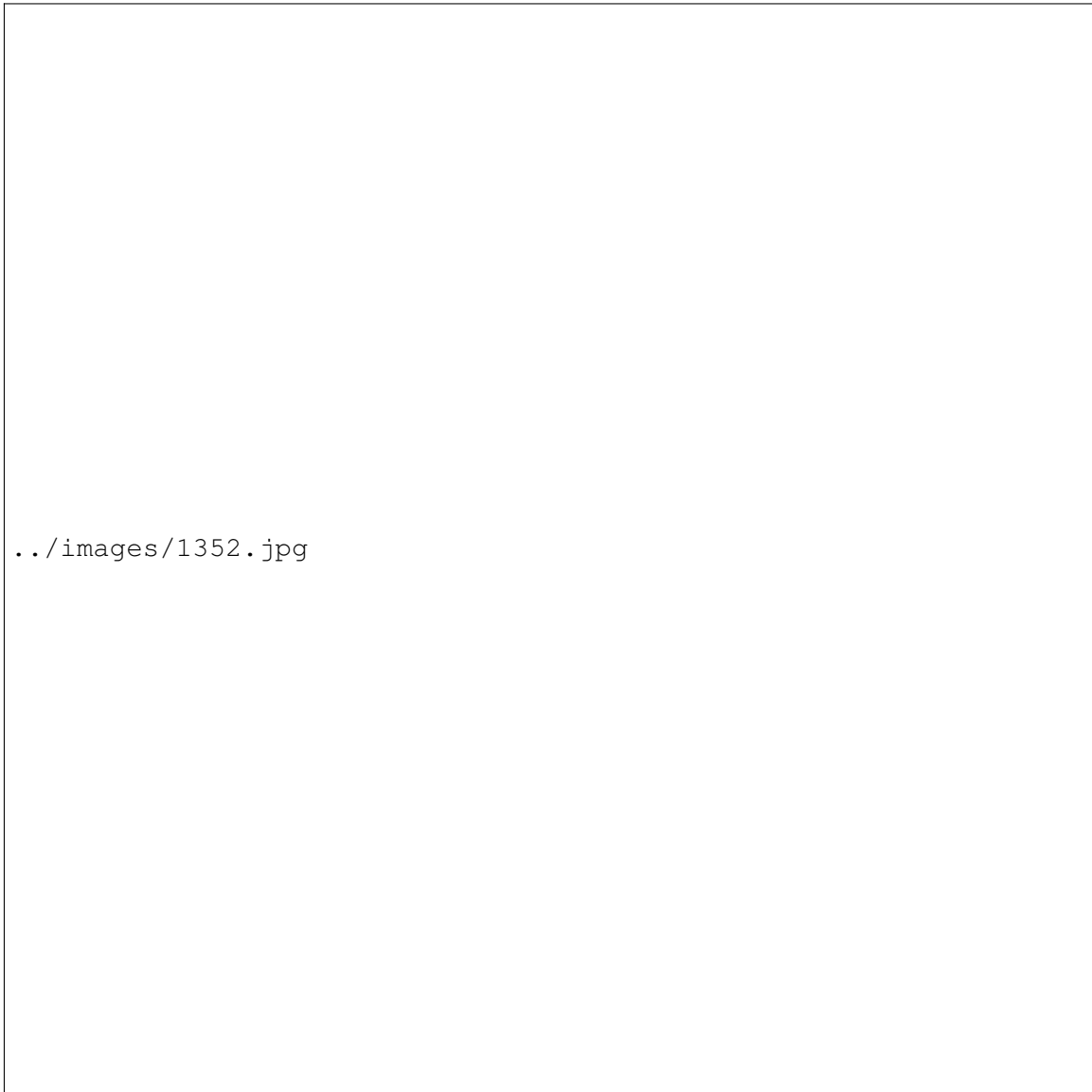


Figure 1352

Now extrude the top of the clayman on the Z axis (in units of 1.0) 7 times so that it looks like this.

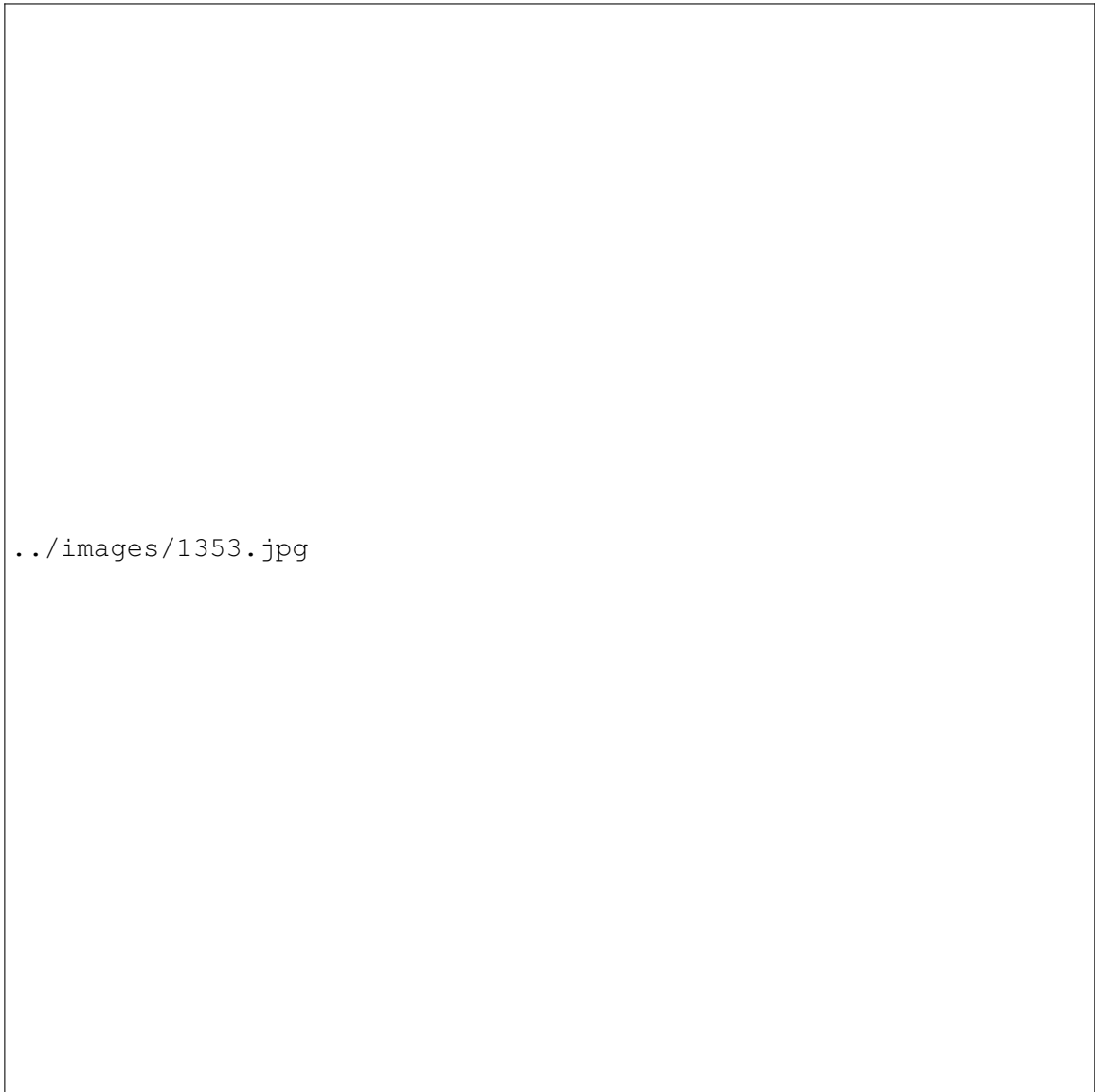


Figure 1353

Now let's make the arms by extruding the 2 boxes that are 3 boxes down from the top. Let's extrude it on the X axis by 1 unit 4 times so that it looks like this.



Figure 1354

Now comes the extrusion of the whole body. As you can see, we only have a flat shape of a rough "Block". so lets move our view around with the NUMPAD arrows and extrude it on the Y axis. Select all vertices with A-KEY. *Extrude > Region* on the Y axis until it reaches 2.0 units. After done correctly, it should look like this.



Figure 1355

With the Clayman successfully extruded, its now time to add some definition to him by adding a subsurface modifier. In the Edit Buttons window, under Modifiers, click *Add Modifier > Subsurf* and set the Levels bar to 2. It should look somewhat like this.



Figure 1356

Now, Select all vertices with A-KEY, and click the Set Smooth button in the Links and Materials tab. Let's get to shaping the Clayman. Using the Border Select Tool (press B to get the Border Select Tool), grab the according vertices to make a shape like this.



Figure 1357

after that's done, select the head vertices and grab them on the Z axis and move them up by 0.2 to make it look like this.



Figure 1358

Now select all vertices with A-KEY, and scale them down (S-KEY) on the Y axis until it reads 0.6000. And finally, select the underside of the foot (the vertices that make it up) and subdivide it to make it more flat on the bottom.

207.3. Adding Materials

Start by going to the Material Buttons window (F5) and Clicking on Add New, and going down to the column of 3 rectangles that have one gray one and two blank white ones. Click on the gray one and in the top-right corner of the pop-up window, you'll see a Hex Code. Click on the current hex code and type in this new hex code to get the right color: 73BCF7. Now hit F6 to go to the Texture buttons window and click on Add New. Change the Texture type to Stucci. Now you'll see a new

tab appear. On it, for NoiseSize enter 0.150. Now go back to the Materials Buttons window. On the far right there will be three tabs in one. On the Map To tab, Deselect Col and select Nor. Now your example should look like this.

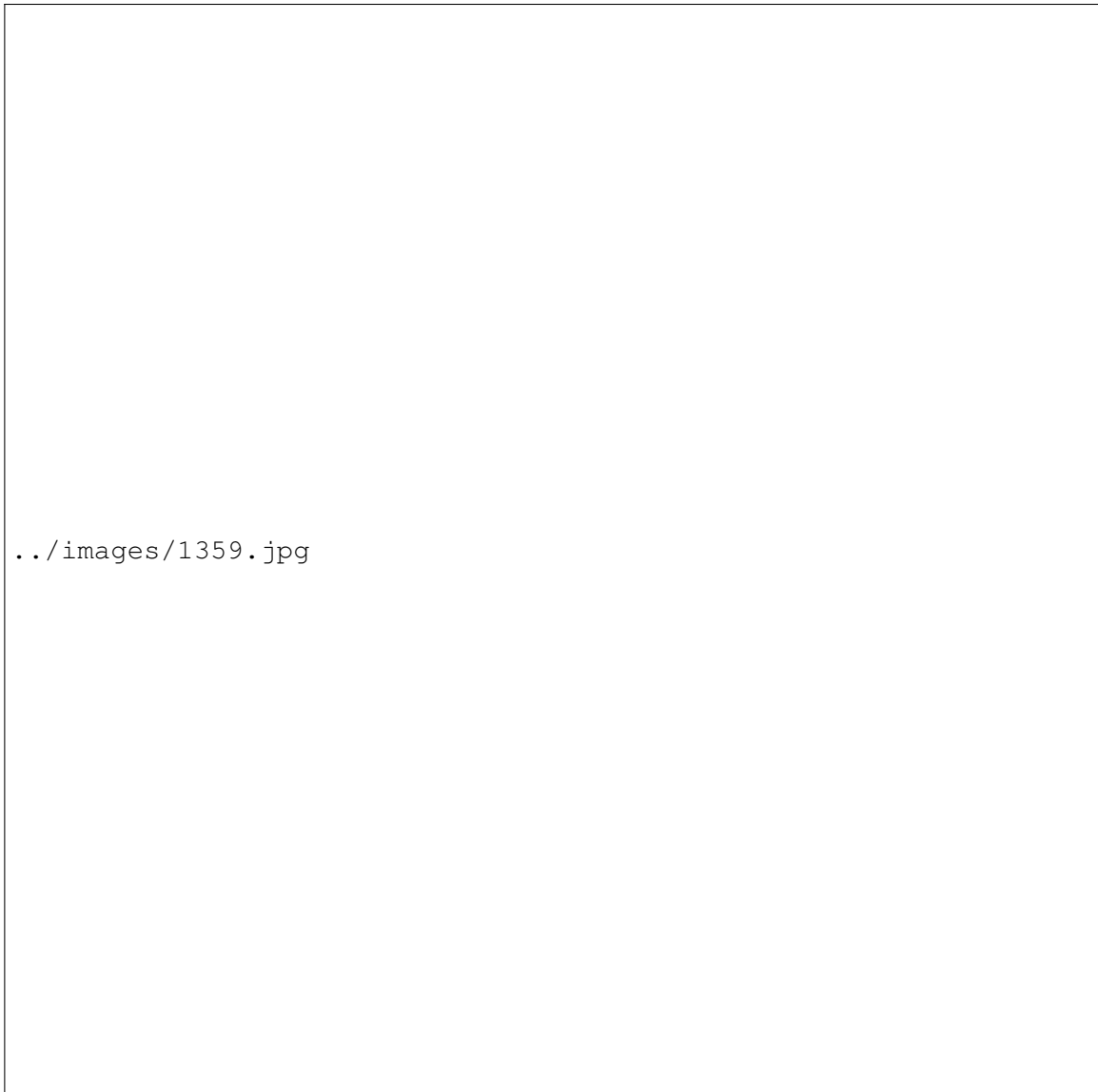


Figure 1359

207.4. Inserting Armatures

Now lets move on to inserting armatures in our clayman. Lets start by going into object mode (alternate between Object and Edit mode with TAB) and adding an armature. Press *Space > Add > Armature*. It automatically puts you in Edit mode upon adding the object. Switch to Object mode and grab it along the Y axis by 1.0 units so that it's inside the clayman. Sure, you can't see it now, but turn on X-axis Mirror Edit and X-Ray in the Edit Buttons window.

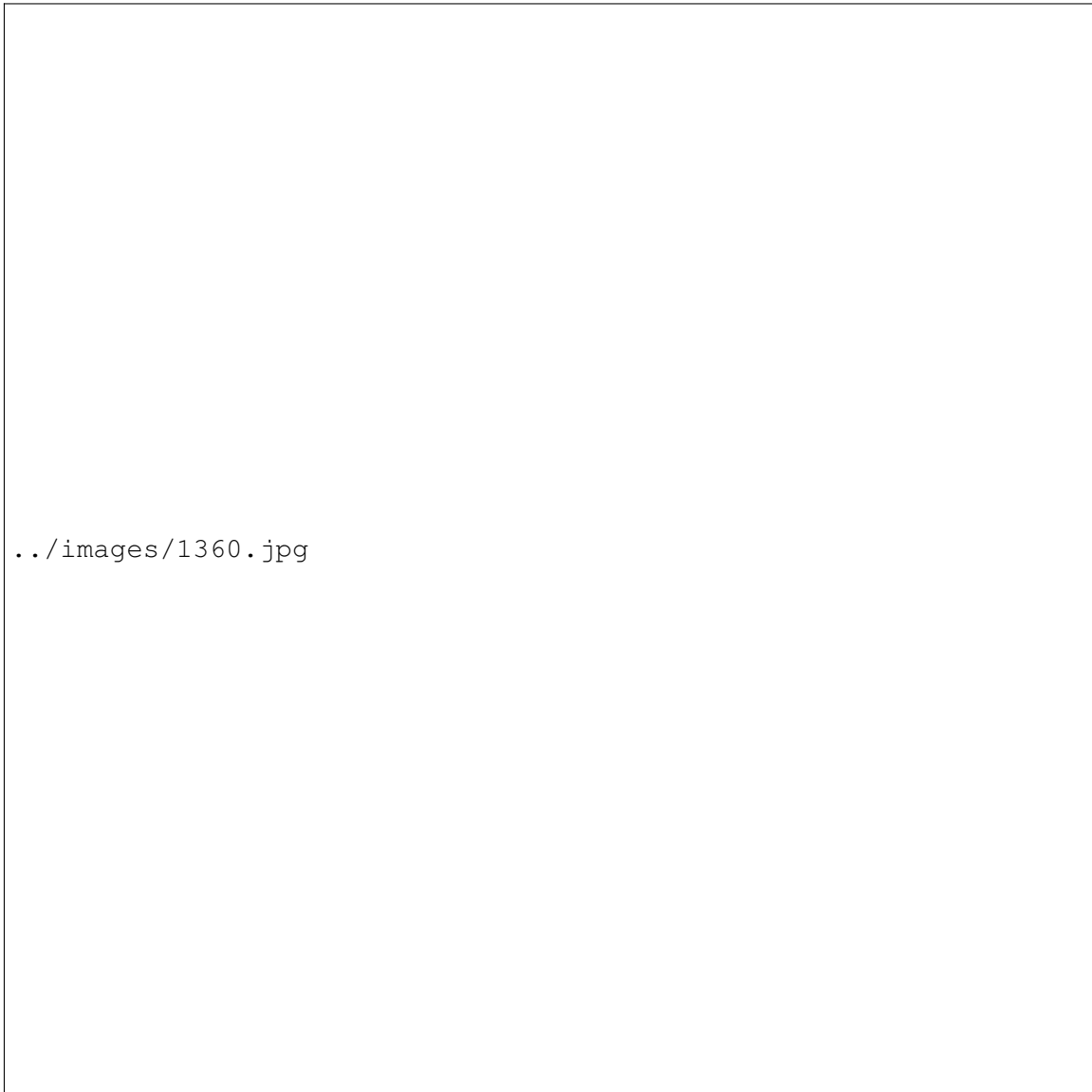


Figure 1360

Now switch back to edit mode, and select the top of the bone. Extrude it on the Z axis by 1.4 units two times so that the tip of the second extrude is right in the center of the two arms. Now extrude it on the X axis by 3.0 units, then grab the tip of it and move it on the Z axis by 0.1700 units. Now extrude it on the X axis by 2.0 units twice. And do the same to the other side. It should look like this.

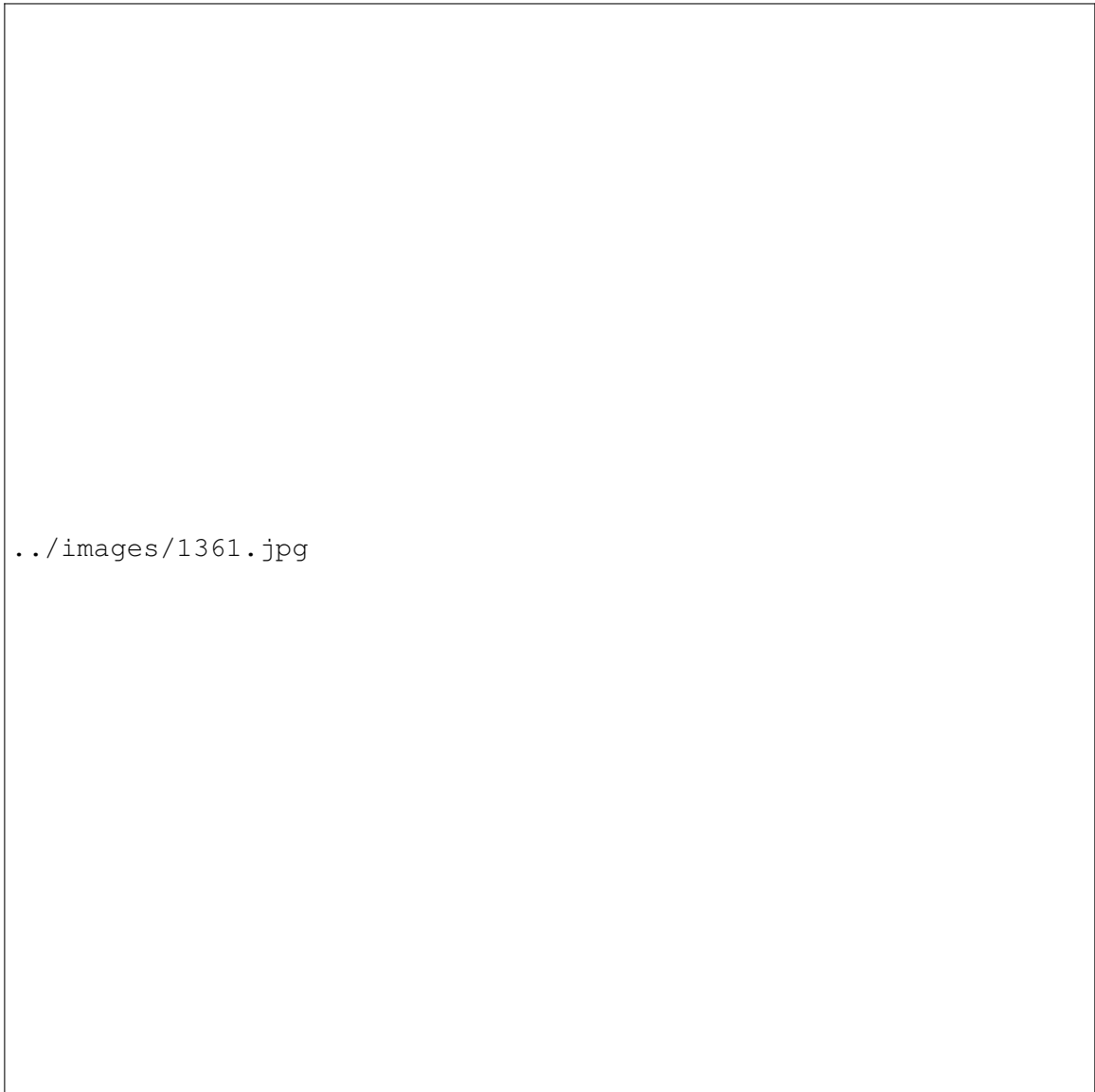


Figure 1361

Now let's extrude the neck. Extrude the top bone along the Z axis by 1.0 units to make a pretty good size neck. Then extrude it up on one side so that the position reads like so. If you can't get it exact then at least try to get it as closest to the number as you can.



Figure 1362

Then do the same to the other side so that it looks like 2 antennae. Now let's extrude the hips and legs. Lets start by extruding the bottom bone by 2.0 units on the X axis and for the Z axis, -1.0 units. Then extrude the foot by -2.0 on the Z axis. And last, select a forearm bone (select a whole bone by clicking the middle of it) and in the Edit Buttons window, you'll see a tab called Armature Bones. Deselect the Con button in that tab, and do the same with the other forearm bone. This allows us to be able to move the forearm out, instead of having it locked to another bone.

207.5. Applying The Armature

Ok, now we're going to apply the armature to the mesh so that when we move the bones, the mesh moves along with them. Lets start by switching to object mode and selecting the clayman mesh,

and under the Modifiers tab, click apply on the Mirror modifier. You may see a pop-up, just click OK. Not only have we just made room in the Modifiers tab, we applied the mirror modifier to the mesh, so now we will have to edit both sides if we want to change something. But worry not, we wont need to change anything from here on. With that new space in the modiifers tab, let's add an Armature modifier. You may see a line that says "OB:" type in the name of the Armature in here (the default name for an armature is "Armature" unless you have made more than one) also, deselect the Envelopes button,for we will not be using envelopes in this tutorial. Here is what you should have.

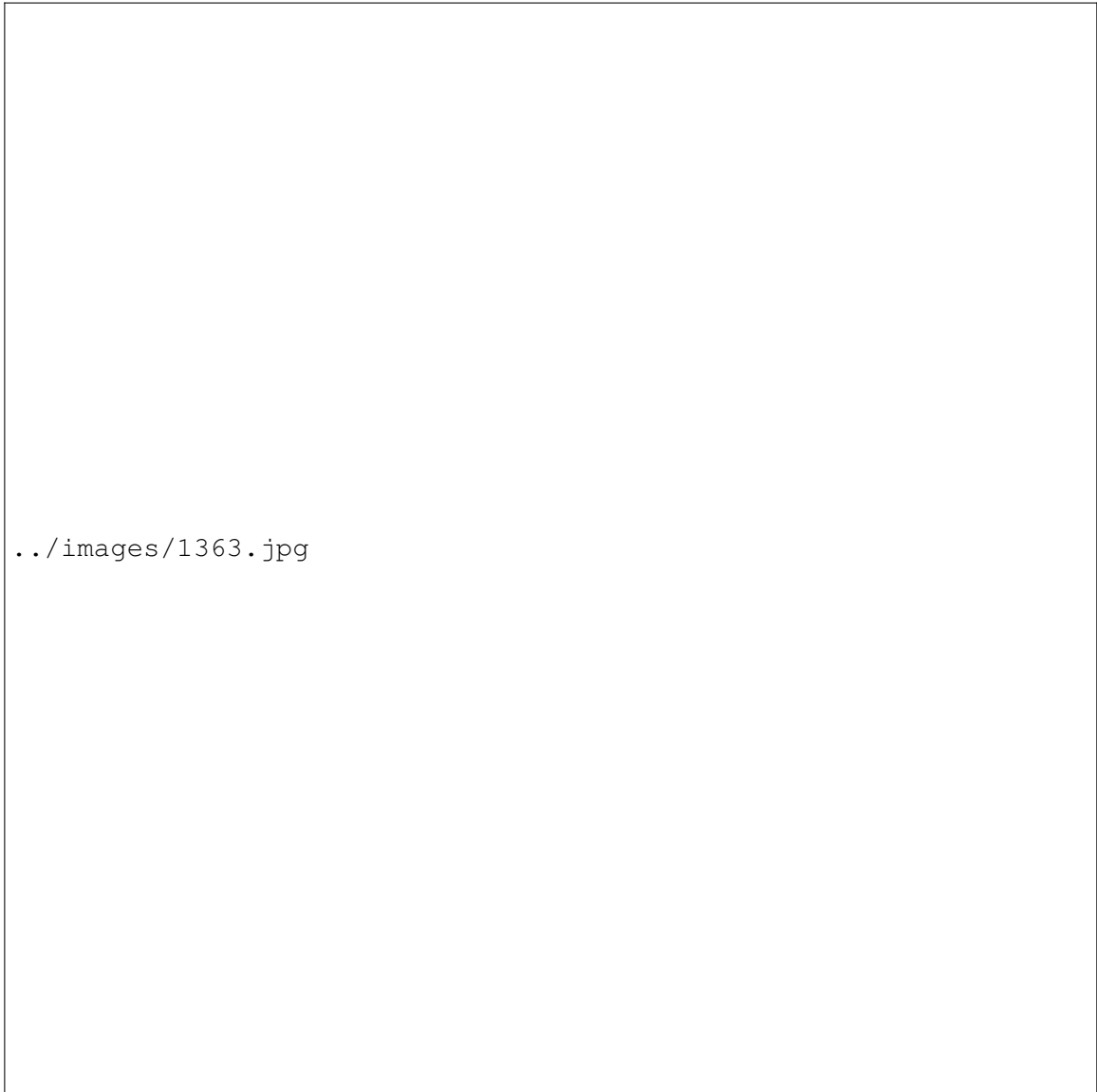


Figure 1363

Now, select the Armature and press CTRL + TAB. If you already have a bone selected, it should be highlighted blue now. We have just switched to Pose Mode. Now, while in Pose Mode, select the clayman mesh (if you cant select it, zoom in closer to get some of the bones out of the way) and press CTRL + TAB on the clayman mesh. We have just entered Weight Paint Mode. This allows us to assign vertices to the bones so that the mesh moves with the bones. Let's start by setting up the

weight painting. You should see a new tab called Paint in your Edit Buttons window. Adjust your settings so that they conform to this.

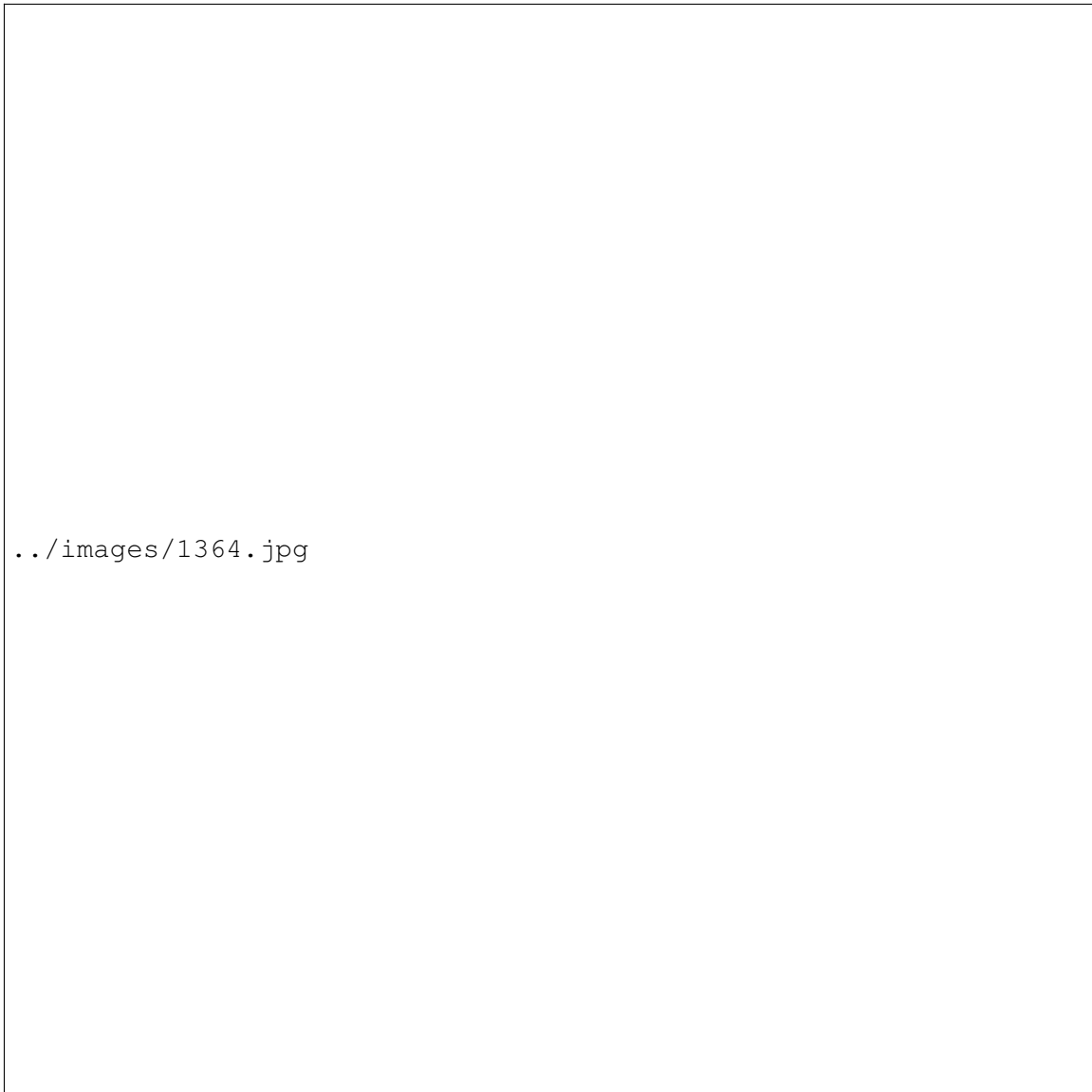


Figure 1364

Start by selecting the top-right bone of the head and start weight painting the front and back (you'll have to switch your view so that you can see the back to get the painting in there) Weight Paint it according to this.

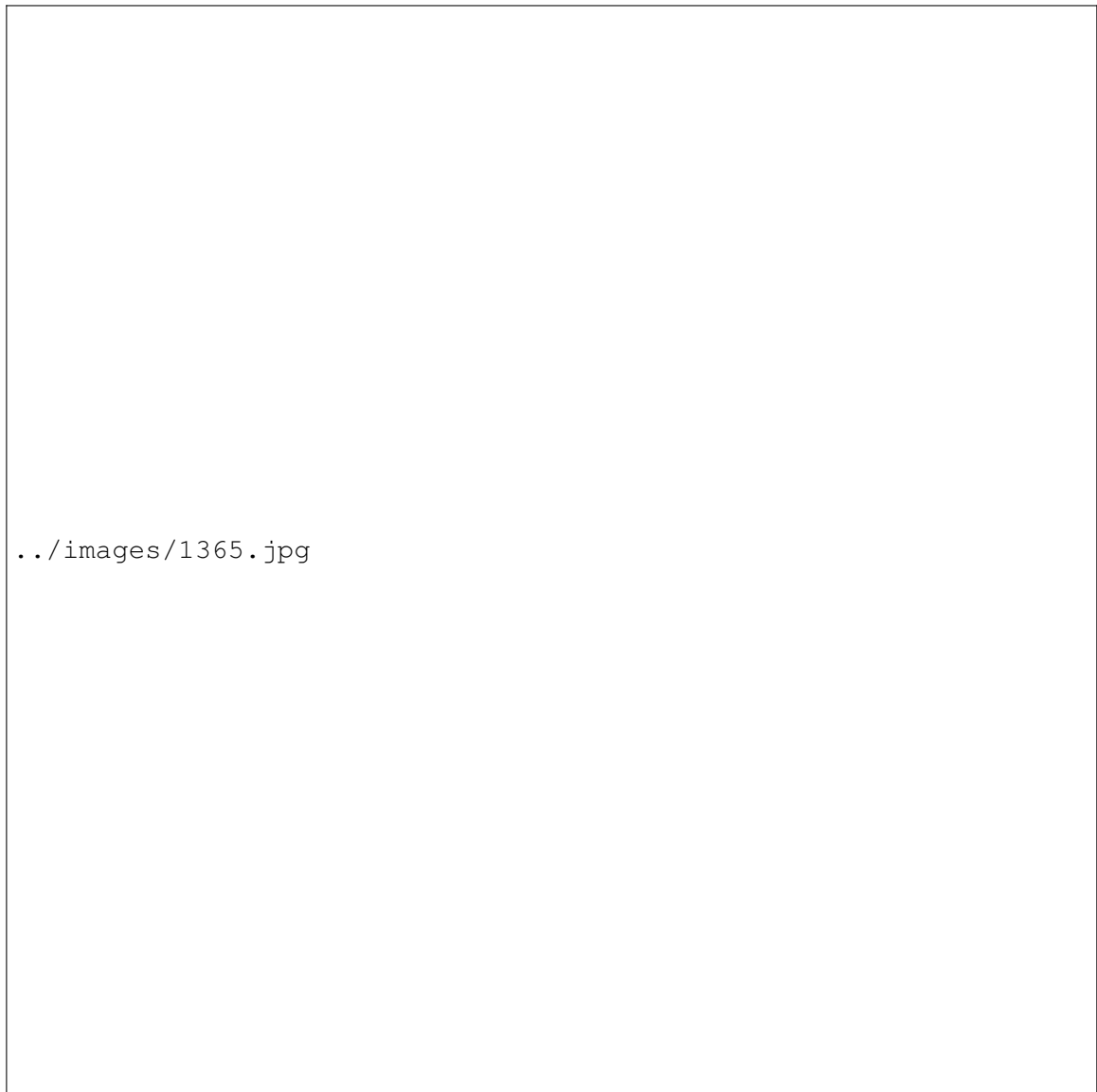


Figure 1365

Now Weight Paint the Neck. (pictures not shown of back view, but be sure to get the back anyway)



Figure 1366

Now the Shoulder.

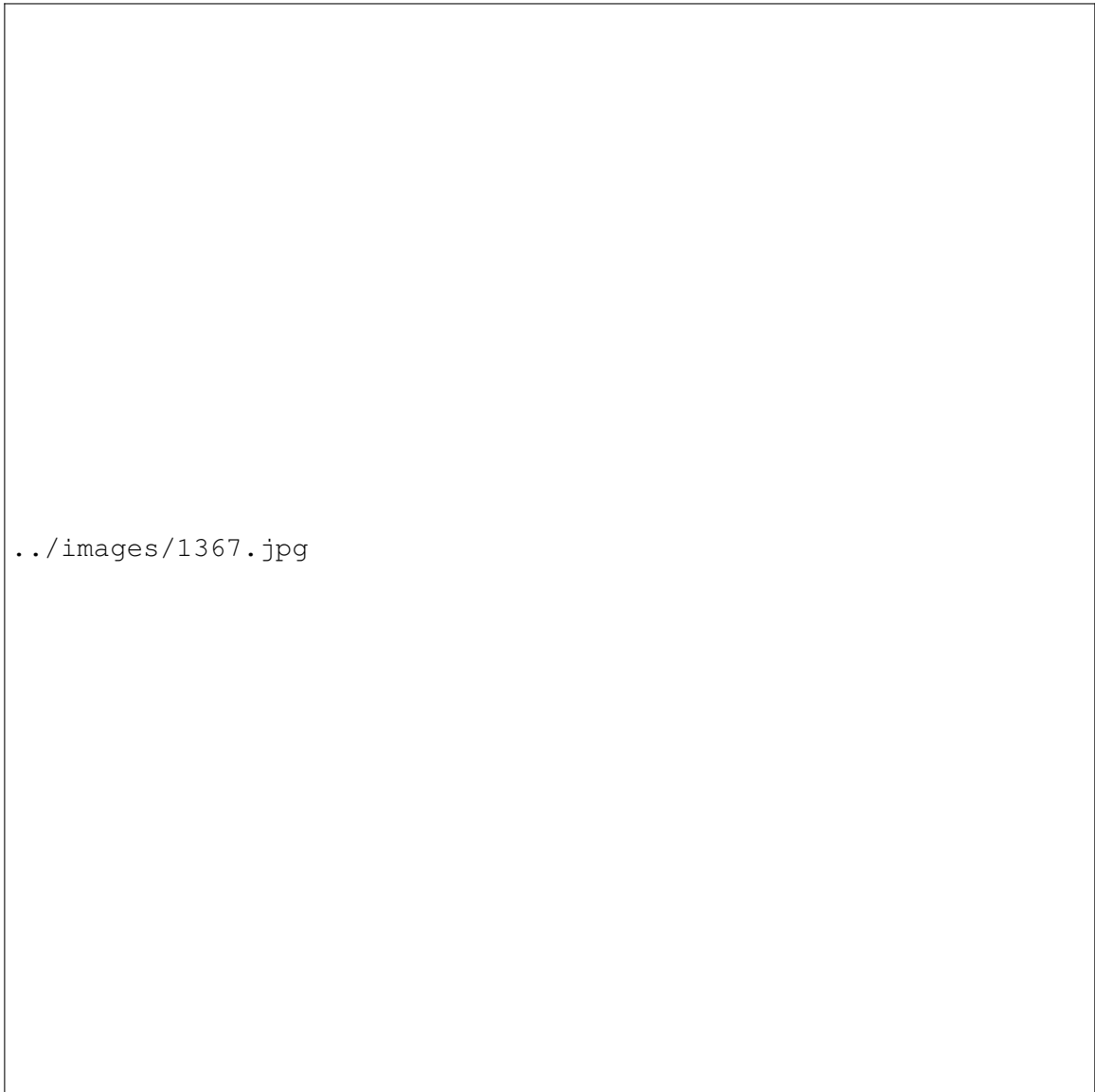


Figure 1367

Now the Forearm.



Figure 1368

Now The Hand.

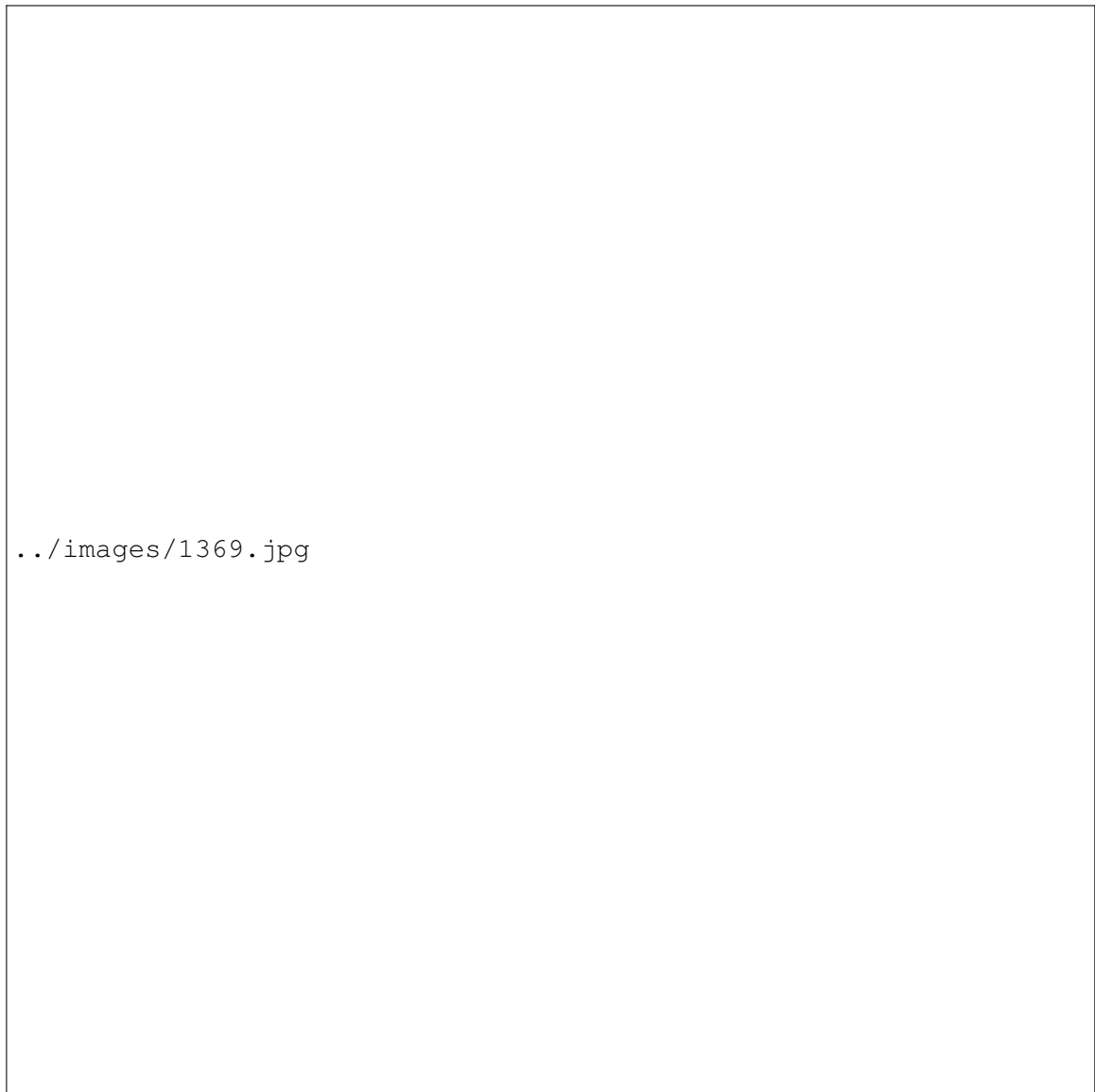


Figure 1369

And after getting the other side of the body, let's get the upper chest.



Figure 1370

Now the middle chest.

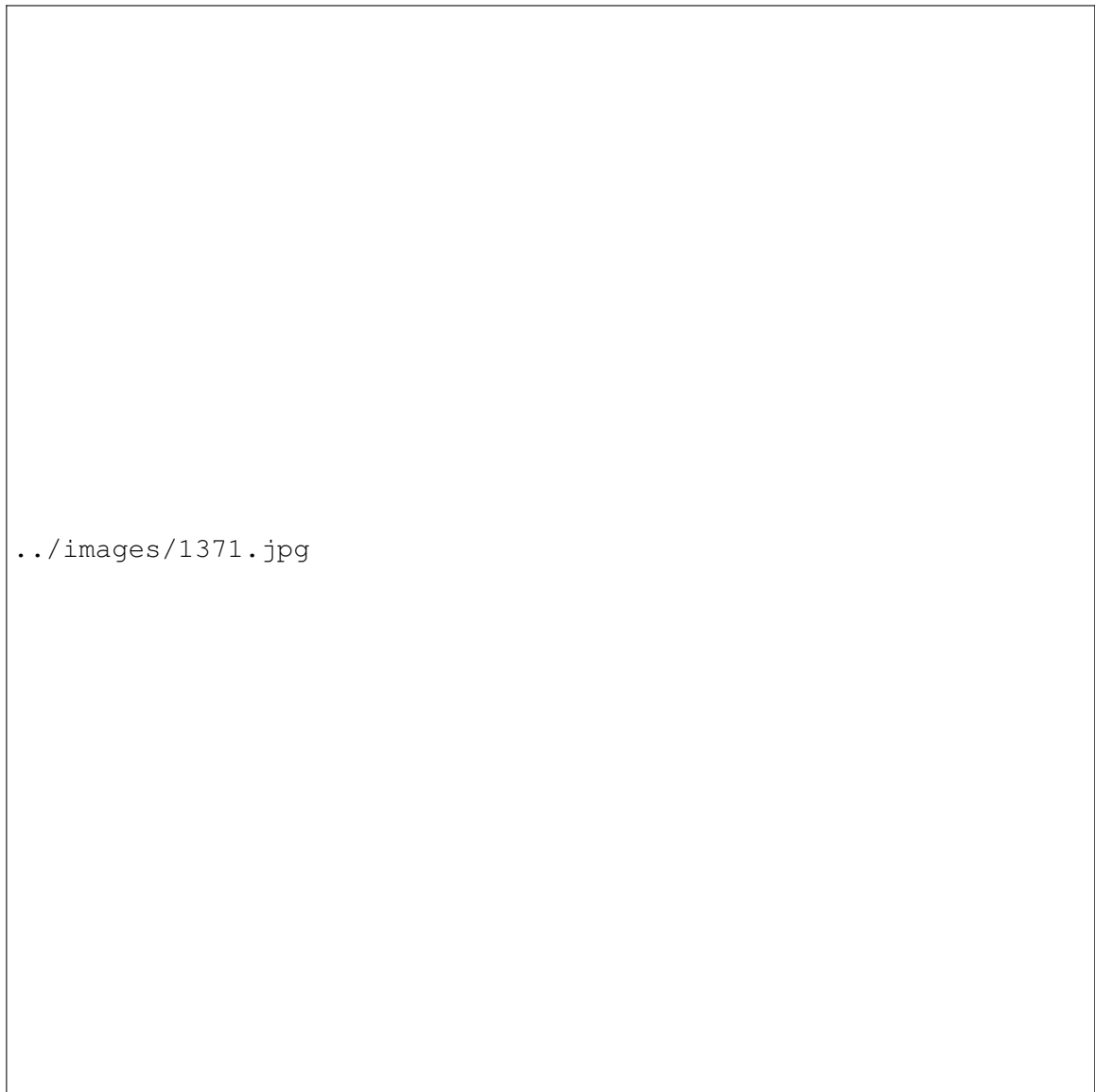


Figure 1371

Now the lower chest.



Figure 1372

Now for the hips.

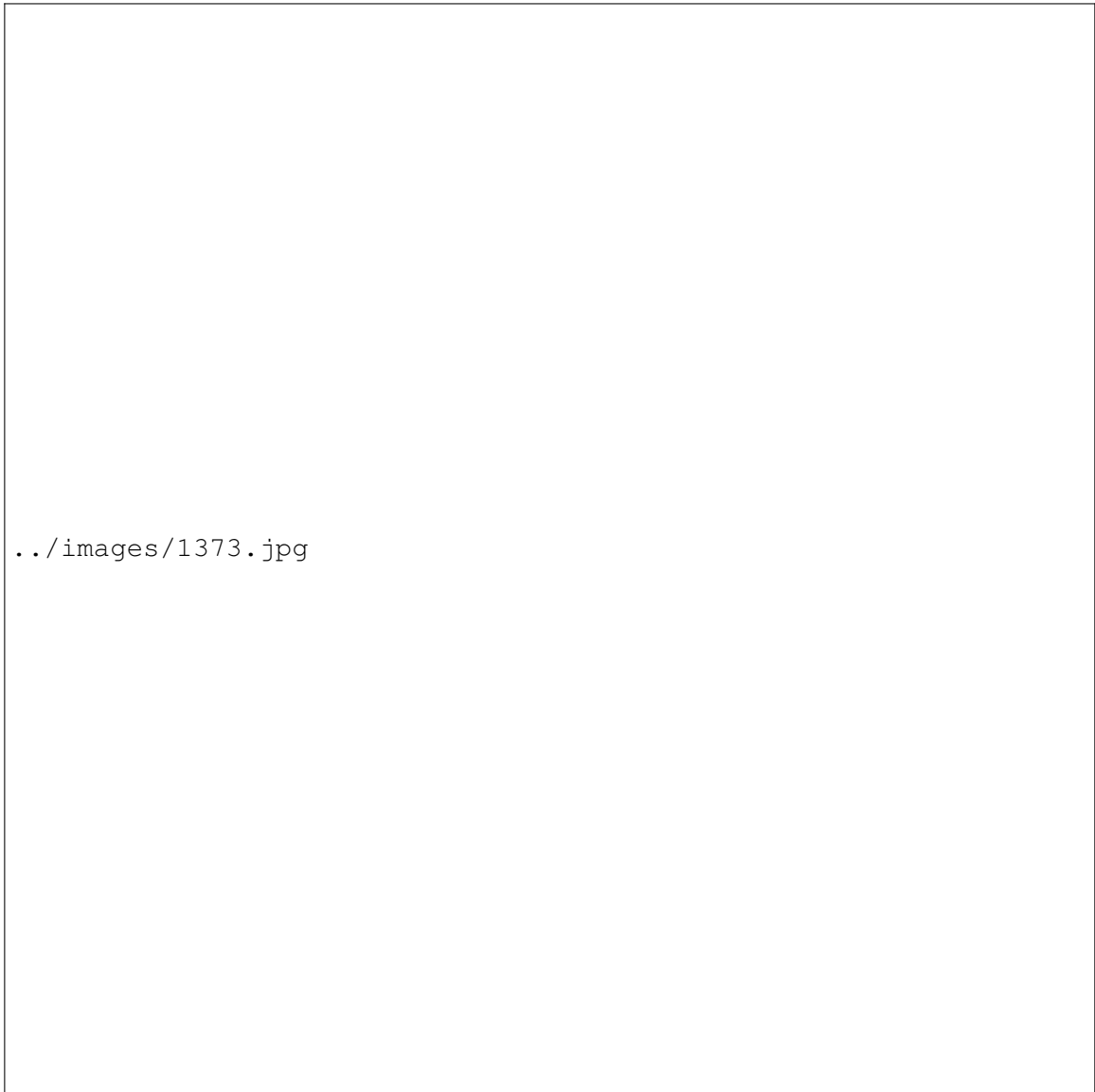


Figure 1373

Now the Feet.



Figure 1374

After doing the other side of the leg, switch back to Object Mode with CTRL + TAB, and select the Armature. You should already be in Pose Mode with the Armature. Now select a bone and grab it to move it. If you have Weight painted correctly, the Clayman mesh should move with the bone. There's two tips to moving the bones around in Pose Mode.

- 1) In the Edit Buttons window, whether Automatic IK is turned on or not affects how the bones move.
- 2) Where your view is affects how the bones rotate. (unless you have it constrained to an axis)

If you find that a part of your clayman isn't moving, just switch back to Weight Paint mode for that bone and repaint it.

208. Organic Modeling

Subsurface modeling

Organic modeling is considered by some the most challenging. Non-organic can mostly be accomplished by extrusion and scaling. Organic modeling on the other hand involves mainly curves, as nature has a thing against straight lines. Because of this, organic modeling is usually done with subsurfaces. To subsurf a mesh, first select it and navigate to the 'Editing' tab. Then go to the modifiers stack and add a new Subsurf modifier.

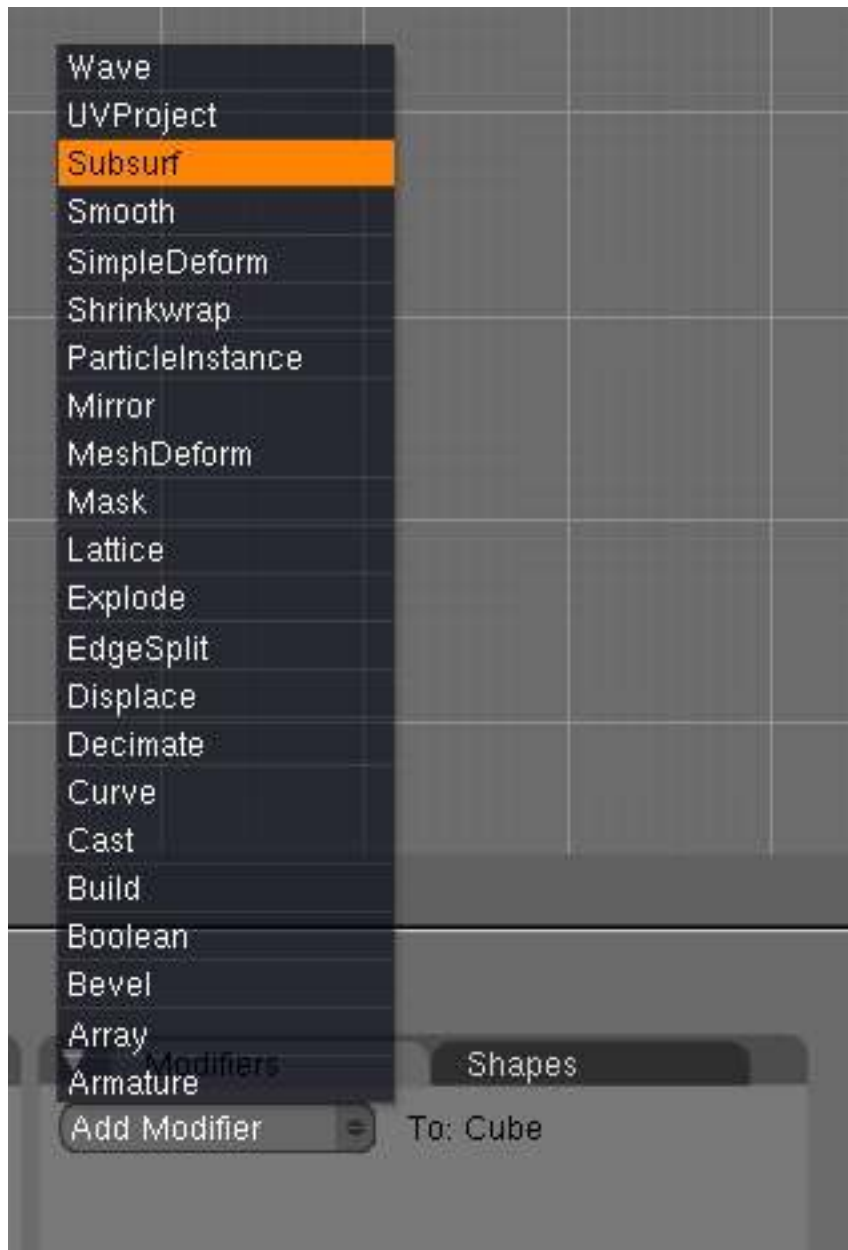


Figure 1375

In Blender 2.5 Alpha 0, Subsurf has been removed from the modifiers list. Instead hit Ctrl, followed by a number which will specify the subsurf level. For example **Ctrl-1** adds a new Subsurf modifier of level 1. This also works in Blender 2.4. Increasing the level greatly increases the number of verts in your model, so make the level relative to the number of vertices in your original model (pre-subsurfed). Subsurf often works best in conjunction with smoothing, so be sure to set your object to smooth, again in the 'Editing' tab, or in 2.5 A0 under 'Object Tools', which you can bring up by hitting the **T** key in the 3D viewport.

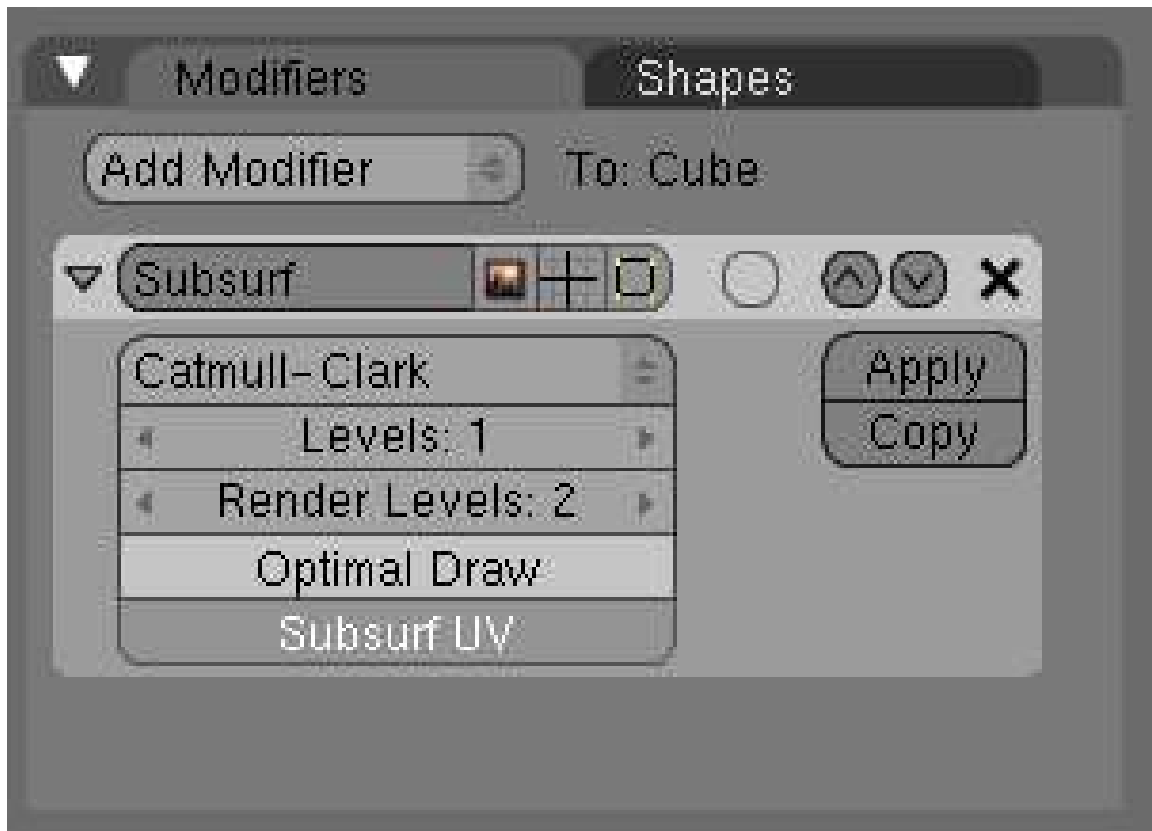


Figure 1376

In this way you can create a smoother, higher-poly shape based on that of the original mesh. This is controlled by the vertices, edges and faces of the original. In order to control the rounding of your mesh you can use two methods: loopcuts, which is a very sloppy method as it adds more verts to your mesh, which serve very little purpose and can get in the way of modeling; and Edge Creasing. The latter can work extremely well in most circumstances. By default all edges are declared uncreased, and so allow complete rounding in subsurfing, by creasing the edges using **Shift-E** you can far better control the amount of rounding on your mesh, and most importantly, without adding extra vertices. However, unfortunately Edge Creasing is not available in Blender 2.5 A0. Yet. And so loopcuts and extra verts are the only option.



Figure 1377 Before

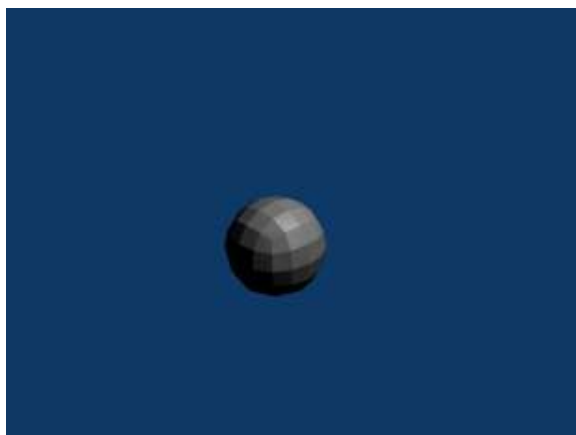


Figure 1378 After

Now, start shaping your mesh into an organic shape. There are various tutorials for modeling bodies and faces, but it is often a good idea to use references. Only use quads - that is, shapes with four vertices. Triangles do not subsurface well, they create bad looking rough surfaces - avoid if at all possible. Also avoid Poles, which are vertices connected to anything other than 2 or 4 other vertices, as they create literal creasing in your subsurfed mesh, that is, sudden bunching up and pinching of faces.

To make the object shape properly, you will need good edgeloops and edgeflow. This means that all your vertices and/or faces line up in a continual line or curve around your model. It is possible to select edgeloops all in one, fantastic when you have to resort to them in controlling rounding, by holding down **Alt** when selecting a vert. This really expands into topology, which is an advanced subject. I would recommend visiting <http://www.subdivisionmodeling.com/forums>,¹ although it has just been closed down, which was a great place to start, however it still (at the time of revision) is active and well worth a look if you are struggling with modeling heads. It is best not to have too many vertices to avoid making the subsurfed shape look rough - in other words, the original shape should be quite low-poly.

¹ <http://www.subdivisionmodeling.com/forums>,

209. Understanding the Fluid Simulator

209.1. Understanding the Fluid Simulator

When I first encountered the Fluid Simulator I had a hard time understanding its behavior, especially the *Start time* and *End time* didn't seem to make any sense at all. Going on a Google spree revealed that many people have problems figuring out the secrets behind fluid simulation, and I didn't find any truly helpful guides. In this little guide I'll try to explain it in a newbie friendly way. It may not be entirely correct, although it might help newbies understanding how it works.

209.2. First of all

Start time and *End time* **are** in seconds. Don't forget this. Even if your simulation seems to go insanely fast when you set *Start time* to 0 and *End time* to 10 and having 250 frames in your animation with 25 frames per second, there is a good reason for this. For now, just remember this, don't let your mind wander and believe that the values are in milliseconds or that you have to do some wicked math dividing/multiplying with frames and so on.

Also notice that the domain is the bounding box for the whole fluid calculation. EVERYTHING is done inside this box. It acts as the floor, ceiling and walls for all of the fluid. This is very important, as the number 1 reason why I couldn't get a good fluid simulation going. If I have time, I will do a section on Fluid > Control, but for now, I will say that it adds a LOT of calculation time. Running on an I7, ATI Radeon 8970 Video, Asus P6X58D Premium, 64-bit Windows 7 and 64-bit Blender, I crashed my computer with the lowest quality settings. So just remember that the domain **MUST** surround the area in which you do calculations. Also note that after you set up your simulation, the domain becomes the actual liquid, so give it proper material and try not to bake until you're pretty sure of your simulation.

209.3. Setting up the scene

We'll learn how the fluid works the practical way. plane

- Start with the default box, this simulation will be very simple.
- Let's work in wireframe mode, press the **ZKEY** to turn off solid mode.
- Go in to camera view by pressing **Numpad0**.
- With the box selected, scale it up to two times by pressing **SKEY**, then **2**, then **ENTER**. This fits the camera fairly well.
- Press **Numpad7** for a top view.

- With the box selected, press **Shift-D** (don't move the mouse or else the duplicated box will move) then press **ENTER** to confirm the duplicated box's location.
- While the new box is selected (and in exactly the same spot as the other box), press **SKEY**, then **.5**, then **ENTER** to scale it to half size.
- Stay away from the mouse, accuracy is important here and I'll explain why later.
- Now we want to move the new box into one of the upper corners. Press the **GKEY**, then press the **XKEY**, type in **-1**, press **ENTER** and the box should move to the left wall of our larger box.
- We want the box in a corner, so press **GKEY Y 1 Enter**, the box should now be in the top left corner from our current view.
- However, we're in 3 dimensions, not 2 so click **Numpad1** for a side view. This time we'll move the box up along the Z-axis: **G Z 1 Enter**.
- Excellent, our setup is done.

209.4. Setting up the simulation

- Make sure you're in Object Mode, and that you followed the above steps precisely.
- Select the smaller box and click **F7** twice. You should get a panel where the rightmost pane says "Fluid Simulation". Click **Enable**.
- Our small box will be the fluid, so just click the **Fluid** button. That's all there is to do with the small box.
- Now select the large box. The Physics panel should still be visible, click **Enable** in the Fluid Simulation pane and then select **Domain**.
- By default your animation should have 250 frames. Rendering should also be set to 25 frames per second by default, this tutorial assumes this setup.
- Since we got 250 frames and 25 frames per second that means our animation is 10 seconds long. So here comes the tricky part, which actually isn't that tricky at all:
 - Start time is by default set to 0 seconds. This means that on the very first frame the simulation has just begun. You could increase this value to say, 1 second and that would mean that on the first frame the fluid simulation has already run for 1 second. We don't want to do this, so keep it at 0 seconds.
 - End time is by default set to 0.3 seconds. What does this mean? This means that on the 250th frame the simulation has run for 0.3 seconds. However, by default our animation is 250 frames long with 25 frames per second, making those 0.3 seconds stretched over 10 seconds. Basically this means that we're watching the show in slow motion, or slightly less than 1/33 the realtime speed. So now you may think "it looks quite realtime to me!", and yes, it does, but why does it do that? Well, that's hard to explain. Consider this: In a world without friction, how far would a drop of water fall after 1 second? The answer is about 4.9 meters. So, if a drop of water falls from 4.9 meters it will take 1 second before it reach the ground. How long would it take the waterdrop to reach the ground if it falls from 3 centimeters? About 0.078 seconds. So why do i mention 3 centimeters? Because by default the size of our domain is 3x3x3 centimeters, or **really** small. If you're like me, you were probably thinking that the fluid was flowing around in a bathtub or a barrel, not in the wrapping of a cupcake. **Set End time to 10 seconds.**
- Since our imagination likes big things, let's crank up that cupcake to say, a swimming pool. Make sure the big box is selected in Object Mode In and look at the Fluid Simulation pane. Just to the left of the "BAKE" button there should be 3 other buttons, possibly named "St", "A", "B". Click **A** for advanced options.

- Some new boxes should appear, Gravity (should be -9.81 for the Z-axis, nothing else), Water and the option we're looking for, "Realworld-size". Also Gridlevels and Compressibility, but let's not care about those now.
- The "Realworld-size" value says how large our domain is in meters, and as you can see it's 0.03 meters by default, or 3 centimeters. We want it huge, so crank it up to 10, which is the limit for Blender 2.45. Now our swimming pool is 10x10x10 meters (don't drown!), remember this because scale matters with fluid. Do not think we're playing with cupcakes again :)
- Now click **BAKE**, and read on while your computer is chewing zeroes and ones.
- Remember how I told you to be very accurate about placing that second box? And how I began talking about gravity, falling waterdrops and stuff? Well, now you're going to see why.
- As stated, our "swimming pool" is 10x10x10 meters. The smaller box we added is exactly half the size (well, in terms of length/width/height, not volume), or 5x5x5 meters. Remember that a drop of water would fall 4.9 meters in 1 second? And that our animation got 25 frames per second? This means that the bottommost part of our blob of water will be exactly 5 meters above the "ground", and that after 25 frames our water should be very close to the ground.
- If you got a fast computer, Blender should be done baking by now. Go to frame 25, for example by using the arrow keys (up/down goes 10 frames forwards/backwards, right/left goes 1 frame forward/backward). Take a close look at the blob, then go forward 1 frame. Notice how the blob hits the ground? Rings a bell, doesn't it? :)
- Although, we're not done! We gotta render our swimming pool. It's easy, but takes time, hit **Ctrl-F12** and go make dinner.
- When the rendering is done, press **Ctrl-F11**, and think of a 10x10 meter large pool. You might want to keep an eye on your kids if your local swimming pool acts this way, though.

209.5. Final notes

Scale matters. It's really difficult to understand fluid dynamics on a very small scale, especially when you don't even know what scale is used. The "Realworld-size" value seems to be left out in many guides, I would recommend you set it to something you can relate to, or you'll end up with simulations that look really slow/fast or having an **End time** value that seemingly makes no sense. Further I'm not a mathematical genius, for all I know I could be way off with my explanation, although this way the values makes sense to me, and I'm able to make fluid simulations without "guessing" on values for **End time**.

209.6. Extra Practice

This YouTube tutorial on fluids might also help: [Link¹](#) and this [Realistic Water Texture²](#)

¹ <http://au.youtube.com/watch?v=yMBx2pYiqnY>

² <http://forum.nystic.com/viewtopic.php?f=34&t=1445>

209.7. Links

- Youtube: Comparison of different values for resolution and real-world size³

³ <http://www.youtube.com/watch?v=okHt4z3FGGM&hd=1>

210. Creating a jewel in Blender

In this tutorial we will create a jewel in Blender. It is fairly simple. I recommend you do this tutorial if you are a noob, because it explains some basic features, but I suggest you read the tutorial syntax and the pages at the very beginning of this Wikibook first.

210.1. Modeling the jewel

Start up Blender if you haven't already. There is a cube in top view (it looks like a square because it is in top view.) Delete it by pressing the X key. Then, to begin modeling, add a circle by pressing **SPACE**→*Add*→*Mesh*→*Circle*. Set the vertices to 10 and make sure it is not filled in. Then press OK.

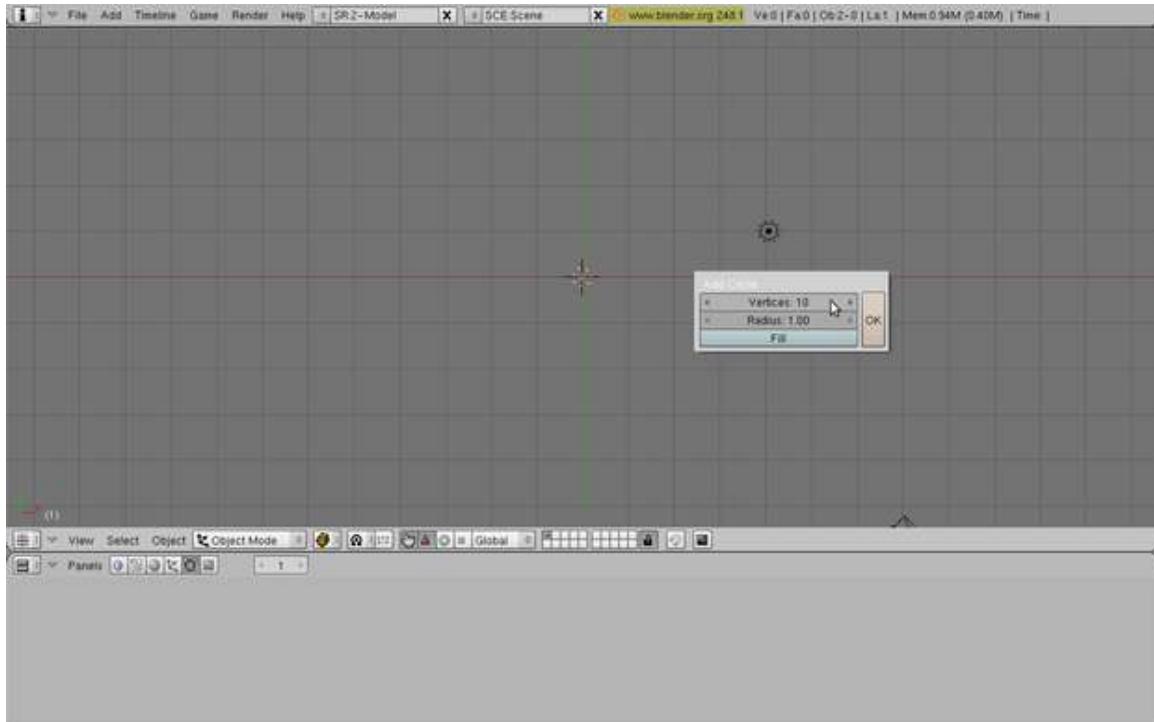


Figure 1379

Next, we will extrude the vertices down. First enter side view by pressing Numpad 3 on your keyboard. As you can see, the circle is flat and not filled in. We will give it some depth. Switch to Edit Mode (TAB) and extrude the circle by pressing the E key and selecting "Only Edges". Move the mouse down and click to confirm the position (you may want to limit to Z axis by pressing the Z key).

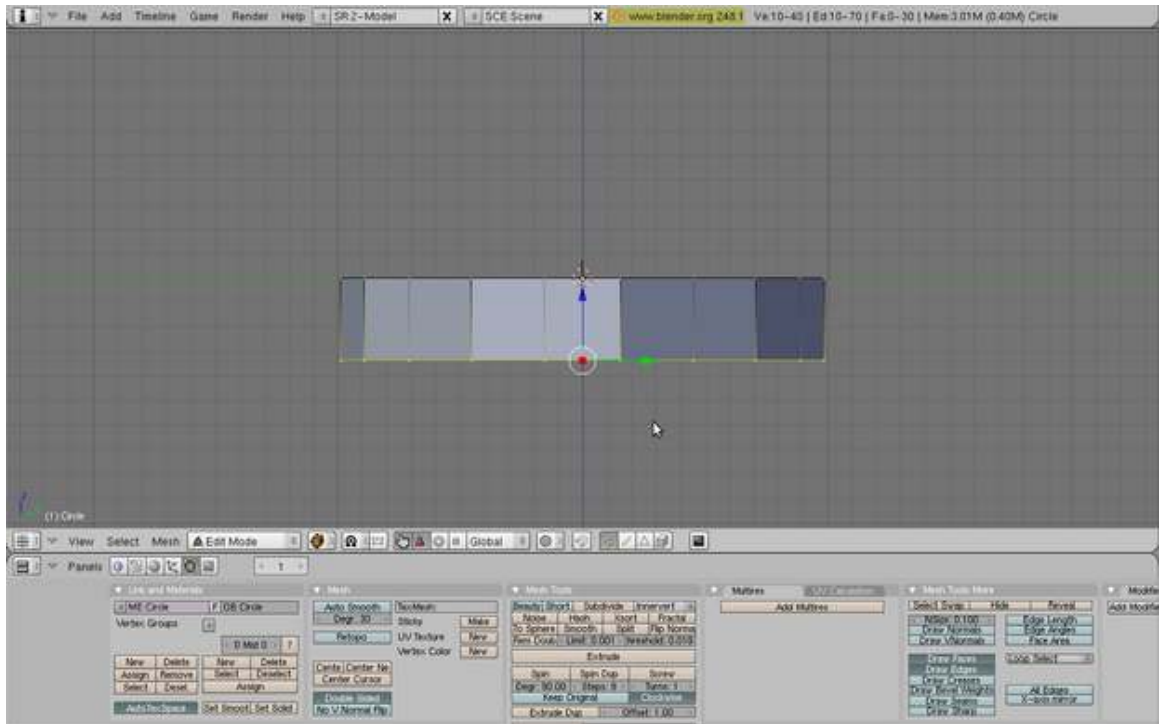


Figure 1380

OK, what we just did is turn the circle into a hollow cylinder by extruding. But we don't want a cylinder. We want the bottom to be a nice tip. To do this, press Alt-M. This creates a Merging menu. Select "At Center". Now the bottom is a nice tip, like we want it!

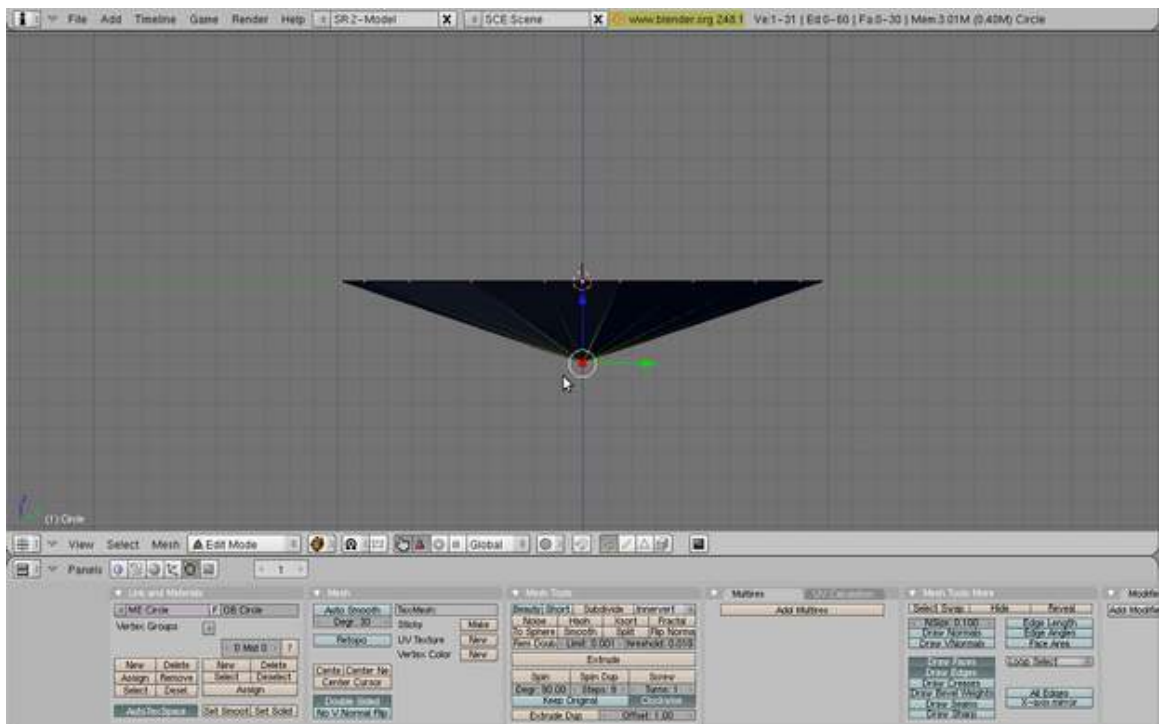


Figure 1381

Now we will edit the top of the jewel. Press A to deselect everything and press the B key. This enters Box-Select Mode. Drag a box over the top vertices to select them.

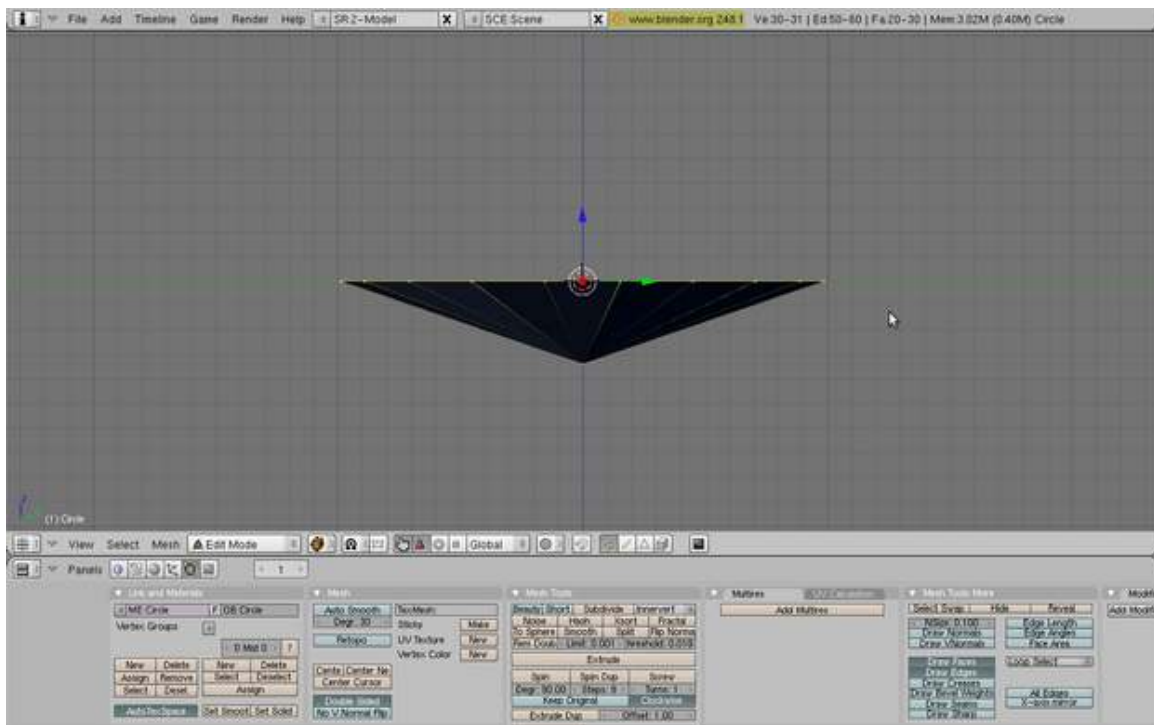


Figure 1382

Now extrude the vertices upwards a bit (again: E key and "Only Edges"). Next, we will scale down these vertices to look a bit more like a jewel. Make sure the ring of vertices are selected and press the S key. This enters Scale Mode. Scale down the top vertices a bit and click to confirm.



Figure 1383

To look around your jewel model use the Middle Mouse Button or Alt-Left Click. Your jewel is looking fantastic! But there's a big gaping hole at the top. We'll fix that. First, enter side view (Numpad 3). Now extrude the topmost ring of vertices but don't move them anywhere with the mouse. If you are feeling doubtful, just press 0 and then Enter (or just click right mouse button). This makes a duplicate of rings placed over the original. Next merge this new ring with Alt-M -> At Center. This merges the duplicate ring, thus filling up your hole.

At last! Your jewel is finished! Feel free to rotate around it and make further adjustments if you are more advanced with using Blender.



Figure 1384

210.2. Shading

Switch back to Object Mode (TAB) and show the Shading Panel (F5). Enable the Material Buttons (click on the button with a red ball). Add new material by clicking on "Add New" button in the "Links and Pipeline" tab. Copy the settings in the screenshot below. If you are having trouble setting the precise value to a slider, just click on the number. This allows a manual type-in. First, make it green by setting R to 0, G to 1, and B to 0. Next set the alpha slider to 0.458 (it looks like an A). Now press "Ray Mirror" (raytracing reflection) to turn it on and set it to 0.13. Now press "Ray Transparency" and set the IOR value to set 1.44. When you're done it should look like this:



Figure 1385

210.3. The finishing touch

To improve the effect, add a plane underneath the jewel. In the Object Mode (TAB) press SPACE>>ADD>>MESH>>PLANE. Scale it 5 times using the S key. Then move it down a bit using the G key (press the Z key to restrict movement to the Z axis).

You may try to render your jewel now. Press the F12 key. You may find that camera doesn't see whole jewel. Move and rotate the camera (using the G and R keys) to set it in the right position. You may want to switch to "Camera View" (Numpad 0) and try out "Camera Fly Mode" (SHIFT F). Try also moving the lamp and see what happens.

I've found that the jewel looks best with Ambient Occlusion on. So go to the Shading window, then the World buttons, click the Amb Occ tab and click the Ambient Occlusion button.

Here is an example of what you can do:

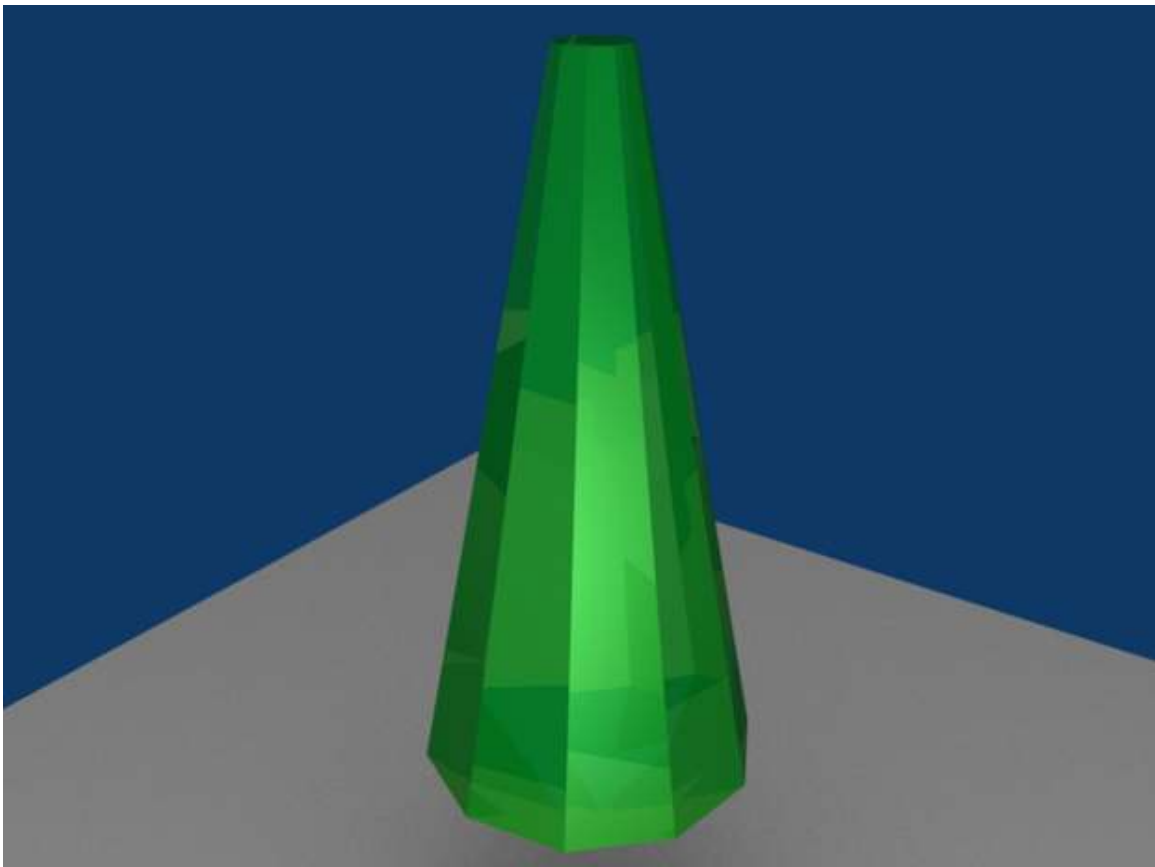


Figure 1386

210.4. External links

- <http://www.blender.org>
- <http://www.blenderartists.org>

211. Modeling a picture

211.1. Modeling a picture

Ever seen an awesome looking picture you wanted to turn into a 3D model? Like a logo or a symbol? Well, it's actually pretty easy... it just takes some time to do.

- First off, you're going to need a picture to trace. I'm currently doing a project for a friend to do with devils and demons, so I chose a demonic looking face for this tutorial: [Demonic Face](#)¹
- Now open Blender and start a new project. Delete the default cube. Before you start tracing the face, you need to set the face as the background image. To do this, click 'view', then 'Background Image'. A box should pop up with only one button in it (Use Background Image), click it. Now some settings appear, we're only interested in one of them for this tutorial. Click the small button with a picture of a miniature folder on it (it looks kind of like a feather pen). It's the first one under the **Use Background Image** button. From there, select the picture you want to trace. Like this: [Background Selection](#)²
- OK, now for the long part. Zoom in to the new background image just a little bit. Now, add a Bézier curve, and size it down a little. Hit **F9** and, in **Curve Tools**, find and click the **Poly** button. Now there should be a few more vertices to work with and the curve should be just a bunch of joined lines. Select one point at a time and using the **GKEY** move it to a point along the background image(or face in this case). Do the same for all of the rest of the vertices, making sure you only have one vertex selected at a time or you'll move more than just the vertex you want to. Once this is done, select one of the end vertices of the curve (it doesn't matter which end) and use **SHIFT+DKEY** to copy that vertex. Move the newly copied vertex to a point along the edge of the face a small ways away from the vertex you copied it from. Continue doing this until you have a complete outline (of the whole face or just one part, like the ear). Here's what it should look like (I did the left ear): [Tracing](#)³. You can't see it in the picture, but six of the points on the right side of the ear are connected, while the rest aren't. In order to get the effect we're looking for here, we need to connect all of the points around the edge to make an outline (make sure not to connect the points across the picture or you'll have a messed up outline).

(user note: hitting CTRL-LMB instead of SHIFT-DKEY will add a vertex that is already connected.)

- To get the outline for the whole face, just do exactly the same thing around all of the edges. We still have a problem though: most of the points aren't joined by a line, so all we have is a bunch of dots. This is easily solvable. Using the **BKEY** or the right click of the mouse, we select a bunch of vertices at a time (somewhere between 5 and 10), and hit the **FKEY** a few times. Every time

1 [http://www.masterpiecepumpkins.com/Graphics/Devil2%20\(1.5\)_____PM.JPG](http://www.masterpiecepumpkins.com/Graphics/Devil2%20(1.5)_____PM.JPG)

2 http://k.domaindlx.com/genis/images/bg_select.jpg

3 <http://k.domaindlx.com/genis/images/points.jpg>

you hit the **FKEY** it should connect two of the points. Do this until all of the selected points are connected, then deselect them and select another group and use **FKEY** to join them. Keep doing this until all of your points are connected. To connect the last two points, select all the points and press the **CKEY**, to close the polygon.

[edit: A better option would be to select a vertex on one of the ends of the whole line, hold down the **CTRL** and left-click on a certain point on the image. This will create a new vertex, immediately connected to the vertex you selected.]

- Now that we've got the entire face traced (or outlined if you want to call it that), we can make it 3D. Hit **F9** again and find the Ext1 and Ext2 properties, shown here: Ext1 & Ext2⁴. Change the values and see what happens. They correspond to the depth of the outline. Try changing them around until you find what looks good. Now, you'll notice that the lines just stick out straight. I'm still investigating how to actually model a head from the outlined face ... so if anyone has any ideas, feel free to add them to this page.
- In order to make it have depth you should make the outline out of mesh points instead of a curve. Add a primitive mesh and delete all the vertices in edit mode, then ctrl click to all point outline. Add depth to the surface in a side view (split views so you can see what you're moving). It helps to have 2 or more reference images, but you can wing it. Usually the final result has to be subsurfed.

(USER EDIT: I accidentally started it with mesh instead of curve. You can do the same thing with extrude, but I have no idea how to go on after that) (USER EDIT LATER: If you subsurf it, it creates a relatively 3D looking image. Its really cool)

(Another user, even later: If you want to turn your curve into a mesh, hit Alt-C while in Object Mode. Note that this is NOT reversible.)

(user edit: You can delete one vertex of a plane, in order to get a line. You may find easier to outline the picture extruding and moving points of the line you created.)

(user edit: you can use this tracing technique to make solid and symmetric models, else, you would really have to use normal modelling)

211.2. Printing a Rendered Image

Render your image. Exit or minimize the "Blender:Render" window. In blender, go to File -> Save Image... Then save your image. Then you can print it as you would print any other picture, using The Gimp, Paint, Microsoft's Photo Editor, or many others.

⁴ <http://k.domaindlx.com/genis/images/make3D.jpg>

212. Modeling with the Spin Tool



Figure 1387 A collection of objects modeled with the spin tool

The Spin tool is a great tool for modeling objects you might make on a lathe quickly and easily.

Lathe objects have circular cross sections along a certain axis. That is to say, when you cut such objects perpendicular to a certain axis you'll get circles. Examples of such objects includes rods, poles, wineglasses, and pails. A Blender render at the left shows some of these.

With the Spin tool you only have to "model" half the outline of your object. The object is completed after you spin this outline. A bit of cleanup here and there and your model is finished.

213. Spin Tool Introduction

213.1. How the Spin Tool Works

The Spin tool works by making copies of the selected vertices in a radial array around an axis indicated by the 3d cursor. Each copies of vertices are connected to previous vertices that corresponds to it with necessary geometry, i.e. edges and planes.

A picture explains this more clearly.

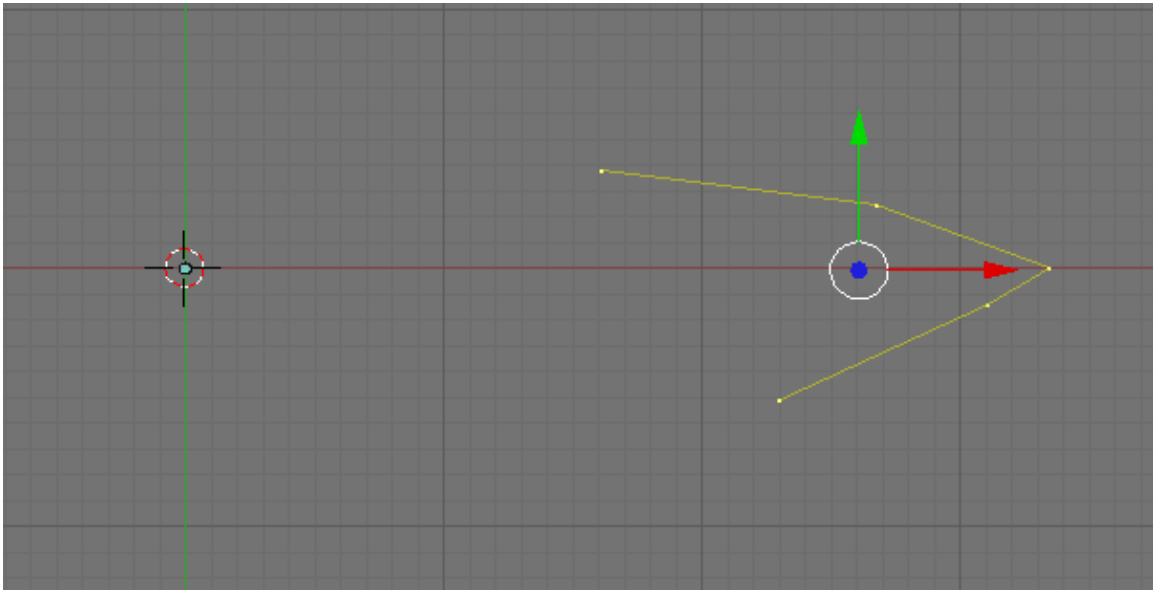


Figure 1388 Selected vertices before using the Spin tool

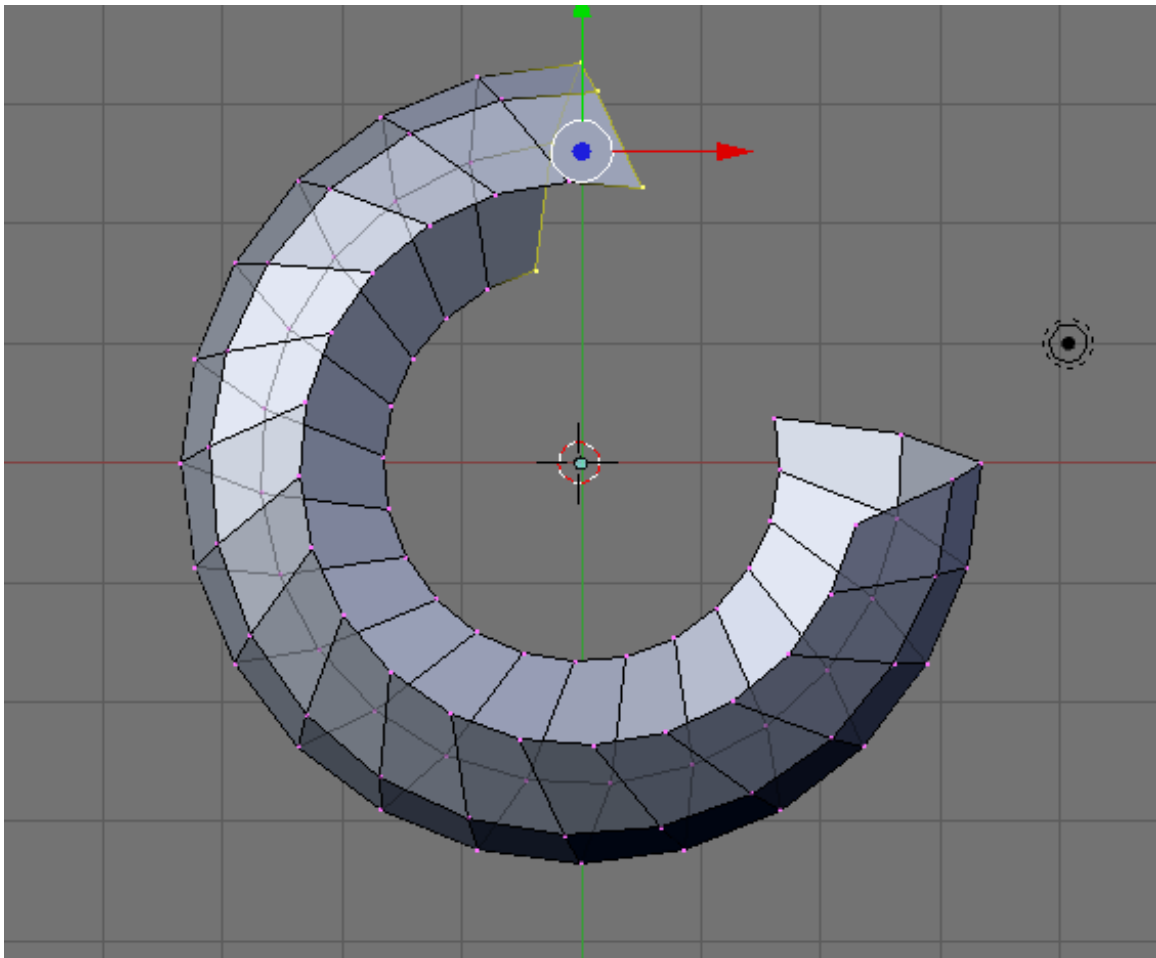


Figure 1389 Result after using the Spin tool

Note that the copied vertices are arranged in a radial array around our 3d cursor and parallel to our view plane.

213.2. Typical Work Flow

- 1) Make half of the outline of our object with connected vertices. If you want to model a wineglass, make the half of the outline of the wineglass.
- 2) Spin our outline. After this the Spin tool would complete our object.
- 3) Clean up the resulting mesh. The mesh created by the Spin tool is not so perfect so we had to "clean" it up.

Modeling Half of the Outline

To start we make half of the outline. Here we do it by creating vertices.

Open Blender. Don't remove the default cube cause we are going to use it. If you don't have any object after you open Blender (like mine), add a mesh object. Any of them will do. I tend to use the plane. At this time it is good to give your object a proper name. Name it sensibly, like "wineglass" if we are modeling a wineglass. This will help us locate it if there comes a time when we have a lot of objects in our scene. We don't want to have all objects to be called cube.001, sphere.003 or something. That could get confusing.

Hit Tab key to enter Edit Mode, if we are not in there yet. Select all vertices by hitting AKey, then delete them with XKey -> Vertices. Then go to the front view by hitting Num1.

Now, we are going to make the outline by creating vertices. Press and hold CtrlKey then click the left mouse button. A new vertex would be created. Click the left mouse button again, another vertex would be created but this time it is connected to the previous vertex by an edge. Click again to add more vertices. We are going to use this to create our outline.

You might notice that the 3d manipulator (red, green and blue arrows) could get in the way. Deactivate it by hitting Ctrl+Space Bar, then select Enable/Disable, or by clicking the white pointing hand icon in the 3d view window.

Lets start making the outline. Delete our previously created vertices first. Using what we have learned, lets create vertices again but this time lets add them to form some sort of outline. let us be creative here. It could be anything you want. Here I make two to illustrate two situations that we might encounter when modeling with the Spin tool. That is modeling objects that are hollow or non hollow.

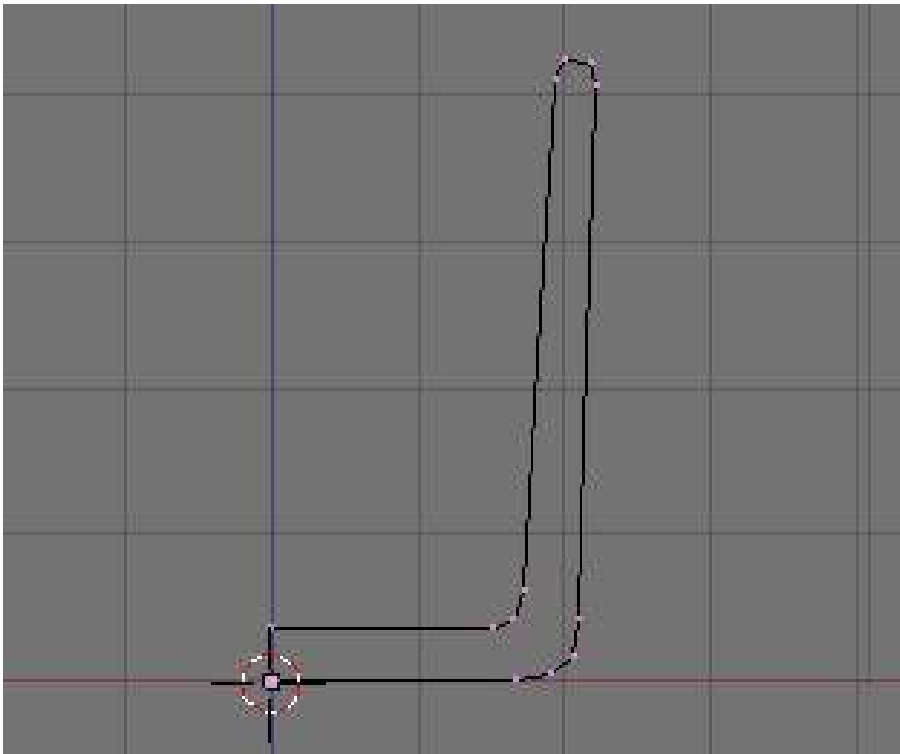


Figure 1390 Half outline of a drinking glass

Here, the first one, is an outline of a glass. Here, we are modeling an object that is not hollow. We had two vertices that are located in the center of our would be glass bottom. We need to align the two central vertices perfectly along the z axis. Select this two vertices then scale it along the x axis to zero (SKey, xKey, 0Key, Enter).

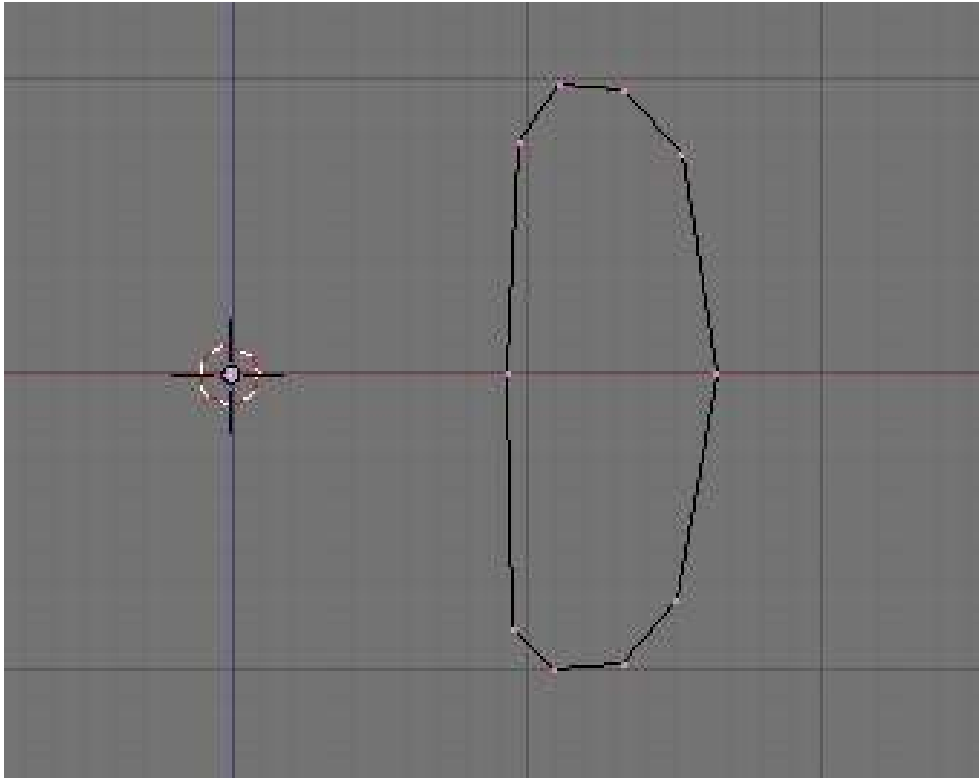


Figure 1391 Half outline of a napkin ring

The second one I will make into a napkin ring. This time our outline is closed and no vertices are located in the center of our would be napkin ring. Here we are modeling an object that is hollow.

To make the closed outline, select the two vertices on the open side then press FKey. The two vertices would be bridged by an edge, thus closing the outline.

Adjust vertices when necessary.

For those who may like to have an image in the background to guide them. Upload a background image by selecting the View menu from the 3d View Window then select "Background Image". A new internal window would appear. Click "Use Background Image" Button, then click "Load". Navigate to where the Image is located select it then click "SELECT IMAGE". Our image would appear on the 3d Window as a background. Bellow, I used a picture of a wineglass.



Figure 1392 Half outline of a wineglass with a background image as a guide

Spinning Our Outline

Now that the outline is finished, we are ready to use it with the Spin tool

Two things you must know is that the Spin tool is dependent on the location of the 3d cursor, and view where we activate our Spin tool.

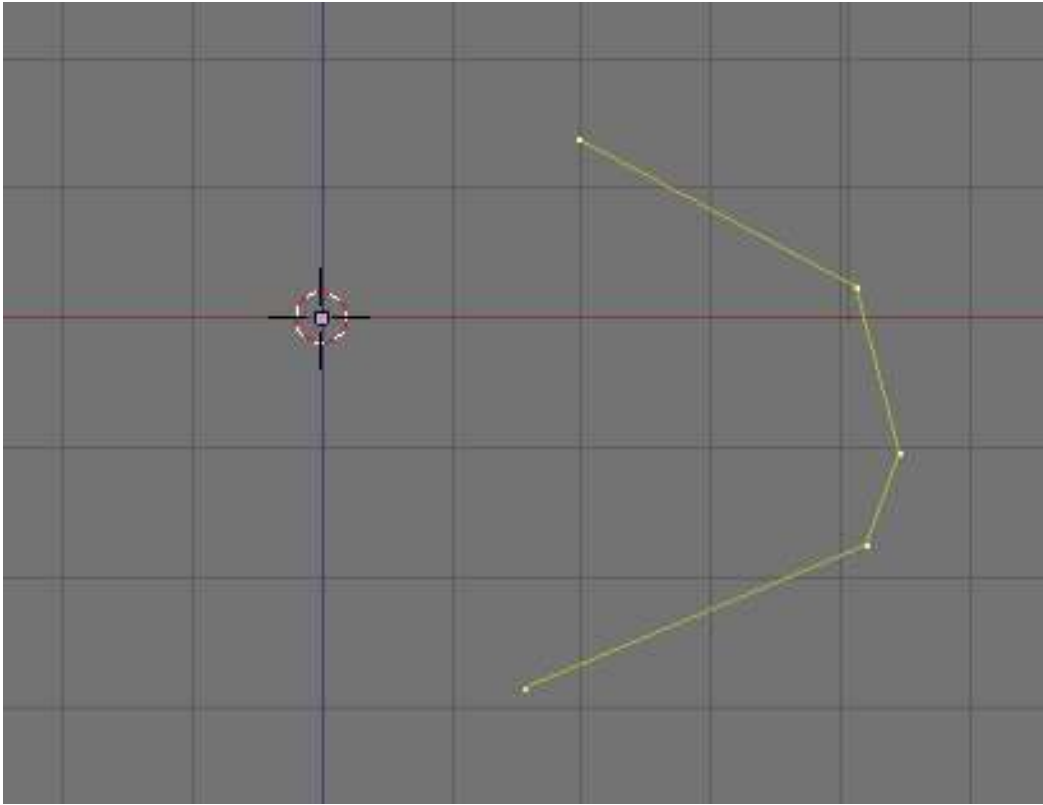


Figure 1393 Setup with an arbitrary shape

Here is our setup with the vertices all selected. In the next pictures are results when this shape is spun with the 3d cursor at different locations.

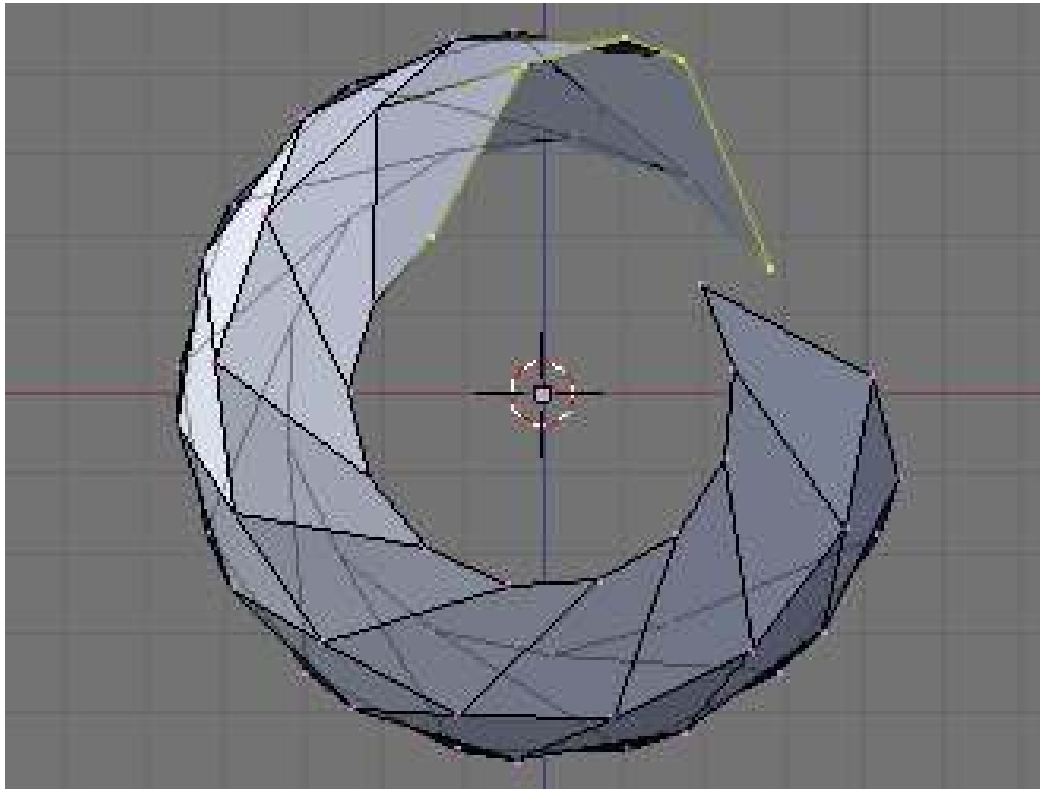


Figure 1394 Shape spun with 3d cursor at the origin

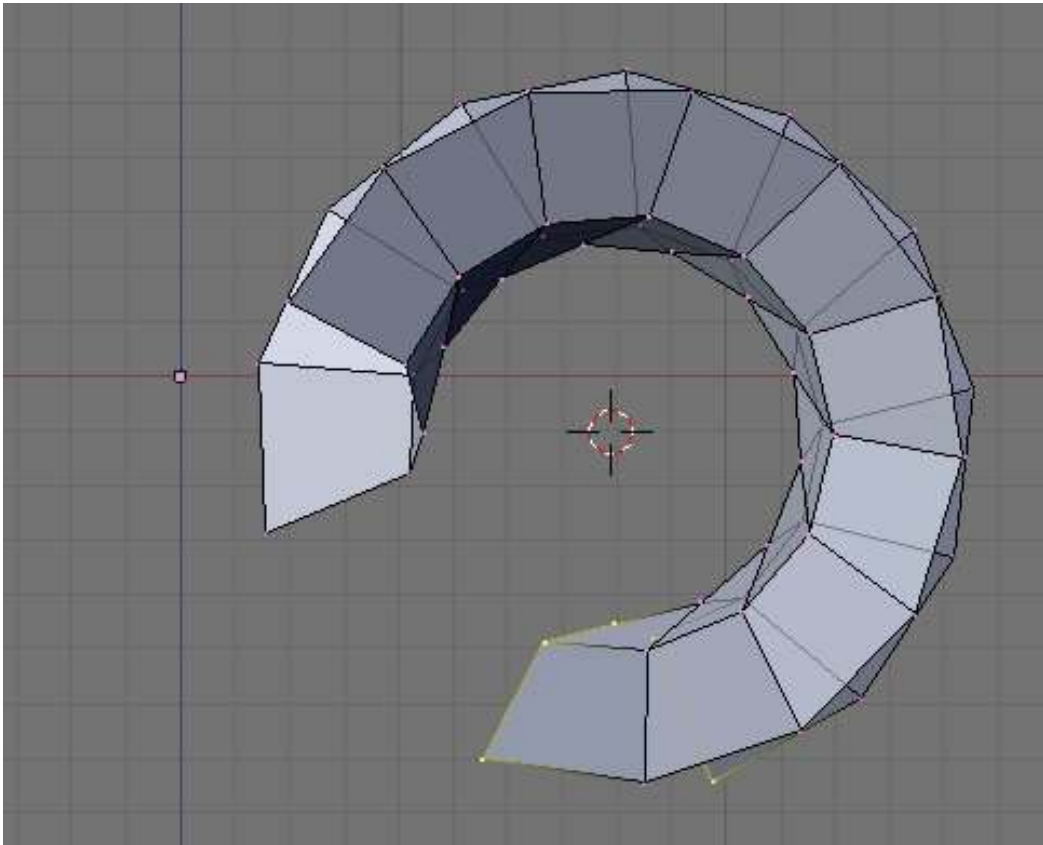


Figure 1395 Shape spun with 3d cursor at another location

Here we have the same shape but is spun around in different views.

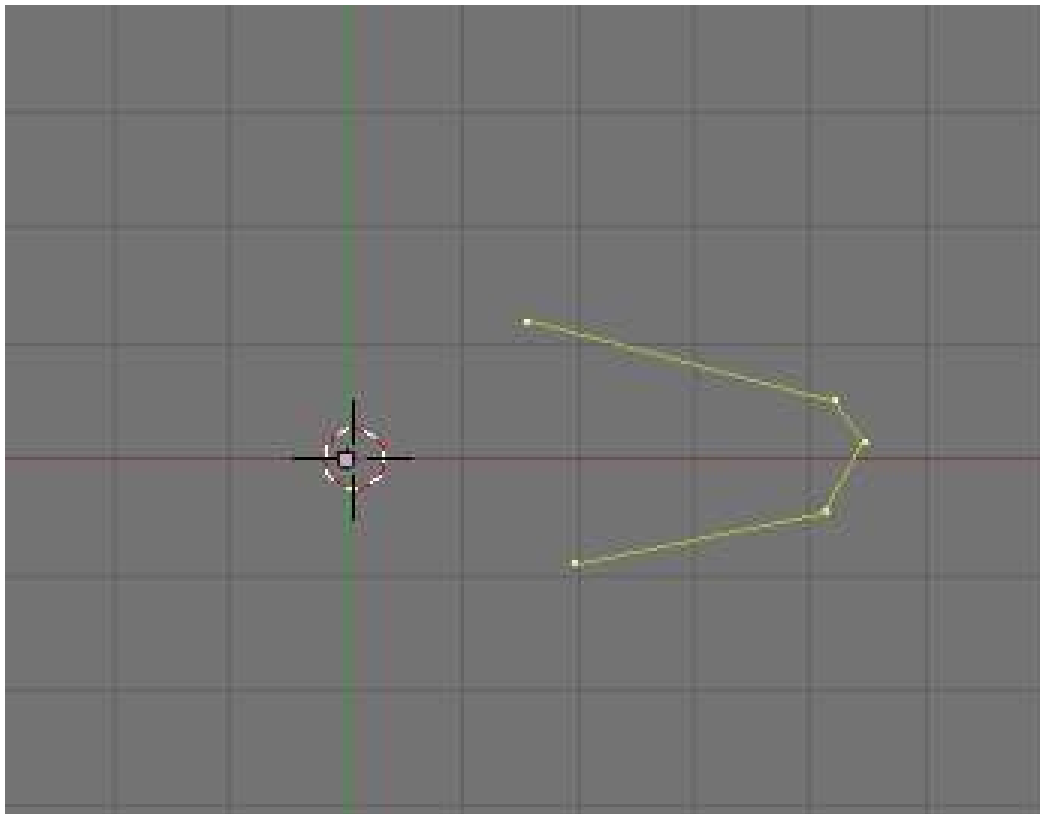


Figure 1396 Shape seen in top view

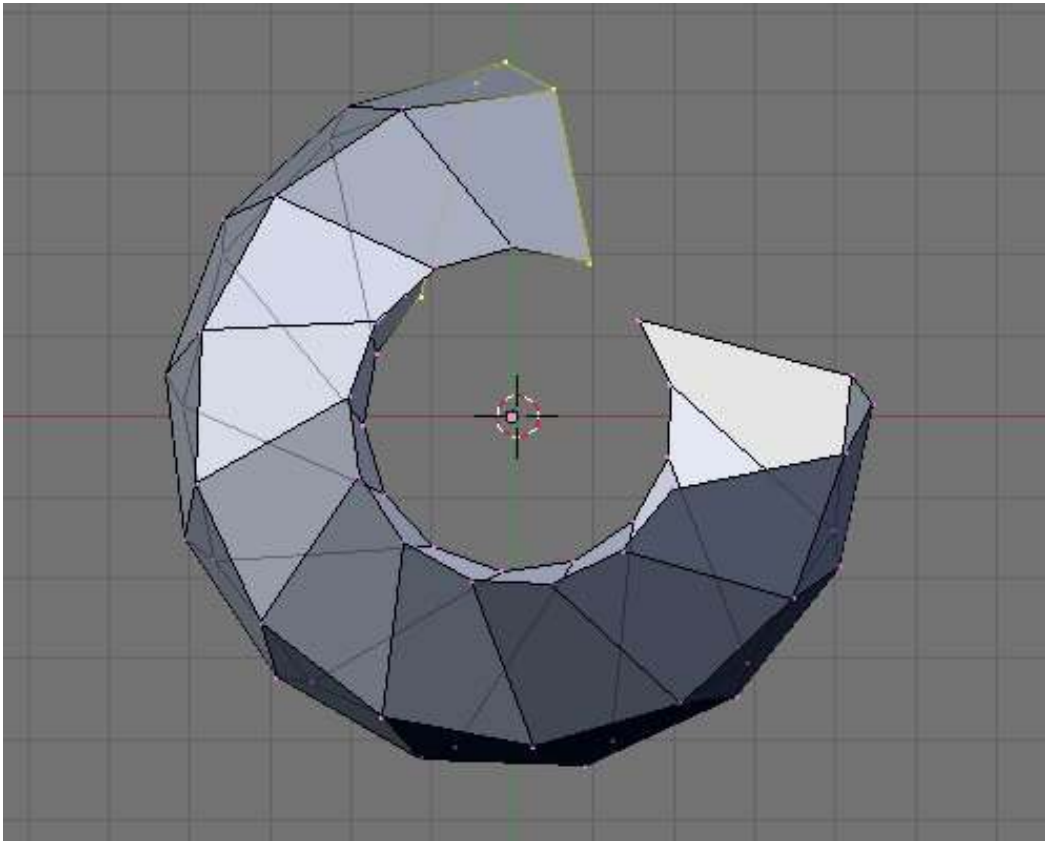


Figure 1397 Shape spun in top view

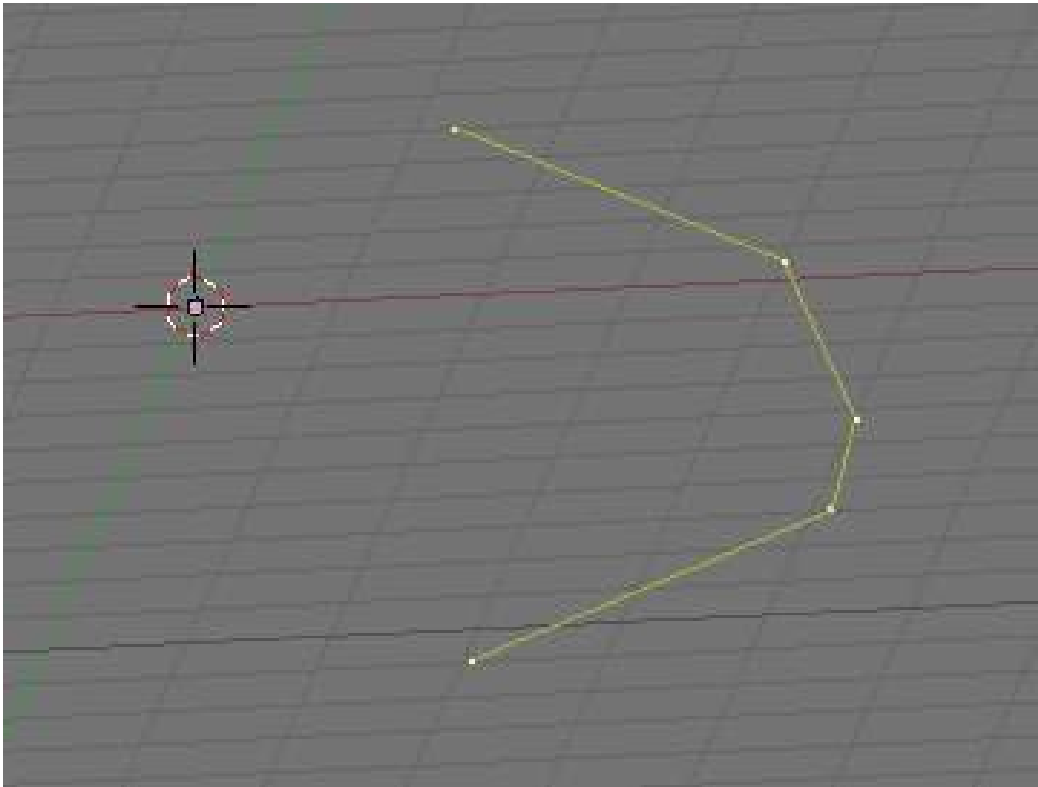


Figure 1398 Shape seen in an arbitrary view

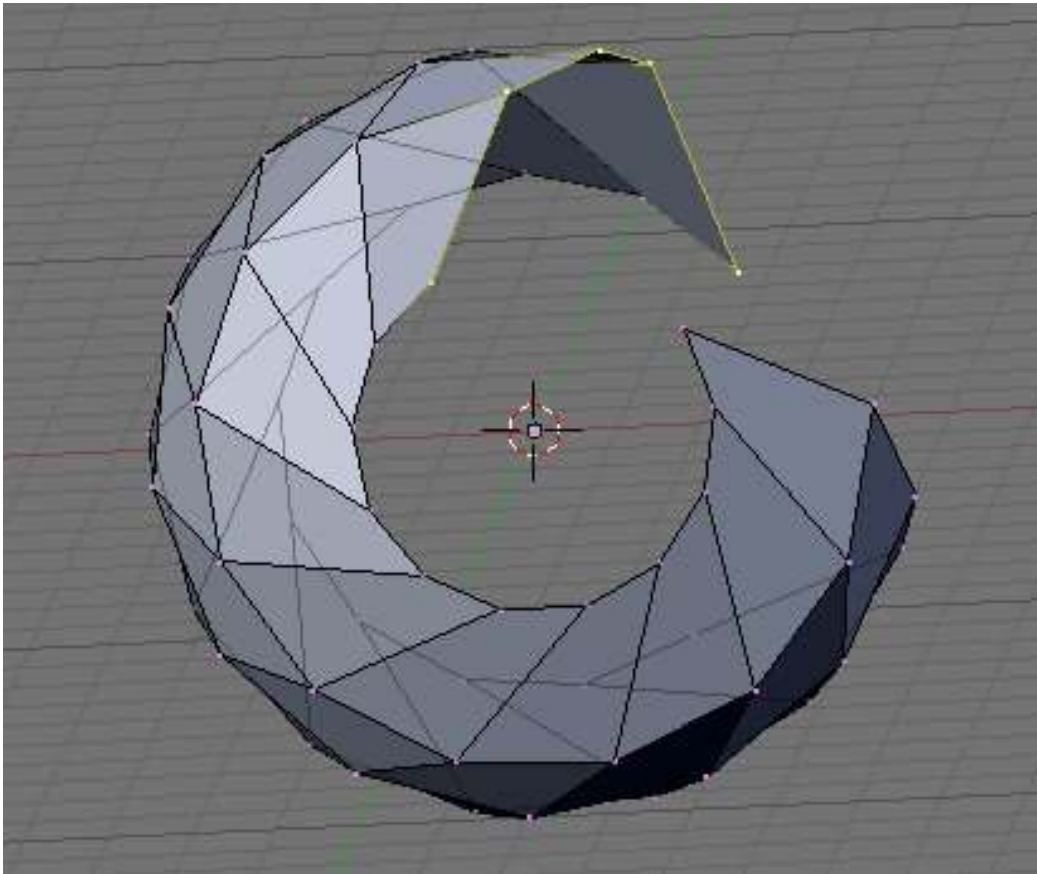


Figure 1399 Shape spun in an arbitrary view

Note that its important only to place the 3d cursor properly relative to the view where you are going to use the Spin tool

Placing Our 3d Cursor

We can see that the location of our 3d cursor is very important in Spin modeling so we need to know how to place it properly. If you prepare it to be, the cursor would already be in position after we make our outline but accidents happen and we accidentally misplaced our 3d cursor. Here are some ways we could make our 3d cursor go to the place where we want it to be.

Snapping

Snapping the 3D cursor is a quick and simple way to place our 3d cursor to its proper location. In the 3d window, hit Shift + SKey. A list of options would appear. We are interested with the last three options, namely the cursor snaps options. They are the Cursor->Selection, Cursor->Grid and Cursor->Active. The name are pretty descriptive but we'll give a short description here.

Cursor->Selection places the cursor to the exact location of our selected element or elements. In case of multiple selected elements, our cursor is placed on the median of the selected elements. In edit

mode, select a vertex and use Cursor->Selection snap to it. Our 3d cursor would jump to the exact place where our selected vertex is located.

Cursor->Selection snapping is especially useful when modeling non hollow objects because there are vertex or vertices located at the center of our model to snap to. Just select one central vertex and snap to it.

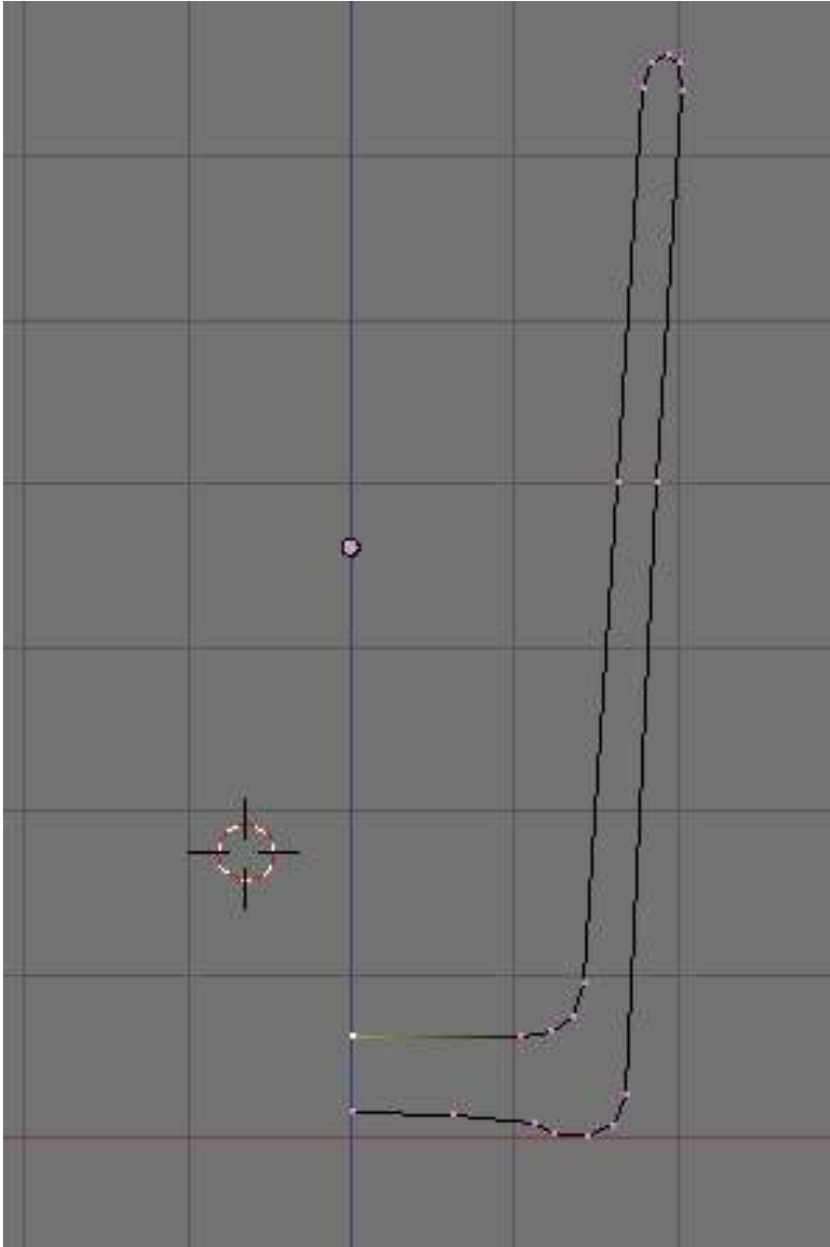


Figure 1400 Here our 3d cursor is not at the center of our drinking glass outline. We select one of the central vertex and do Cursor->Selection snapping.

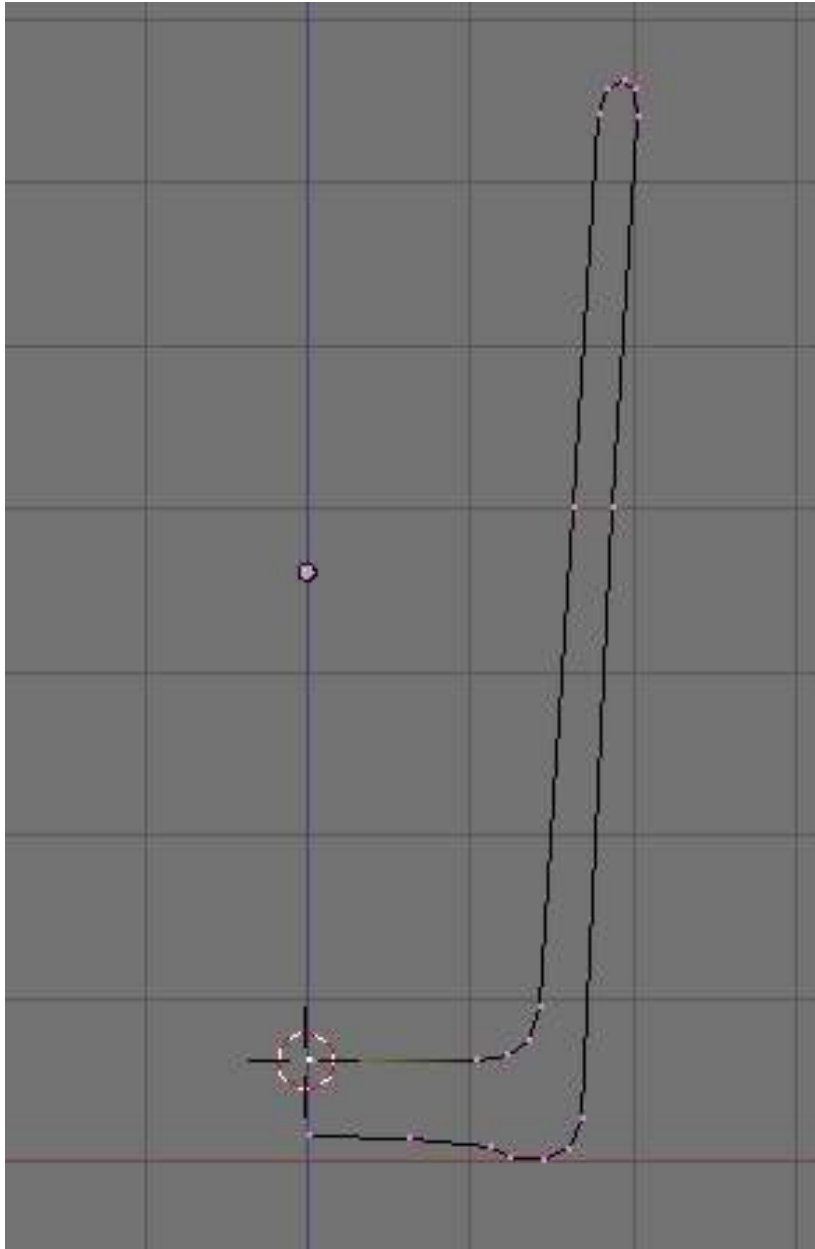


Figure 1401 Result after doing the Cursor->Selection snap.

Cursor->Grid snaps the cursor to the nearest cross of the visible grid. Try clicking at a center of a square in our Blender's grid and use it. See its effect.

Cursor->Grid snapping is useful for spin modeling hollow models. Usually if you are careful, our object outline would be on a plane that contains one of the axes. If we make our outline in front view as advised, go to the top view (or side view depending on what we are modeling). If you are doing okay, you will see all our vertices aligned perfectly along the x axis. It just a matter of LMB close to a cross on the grid that contains our objects center then snap to it.

Cursor->Active might not be so helpful in spin modeling. Cursor->Active snaps the cursor to the active element (usually the element that is last to be selected). Try it by selecting several vertices,

then do Cursor->Active snap. The vertex where the 3d cursor goes is the active element in your selection.

Spin Tool Parameters

The spin accepts two parameters to execute its function. These are Degrees (degr) and Steps.



Figure 1402

Degrees specifies the angle at which the selected geometry will be rotated from its initial position around the axis. It accepts values in degrees. A 360 degrees value being a whole turn around the axis. Values lesser than 360 produces something like a wedge is sliced from the side of the form.

p Sample Outline

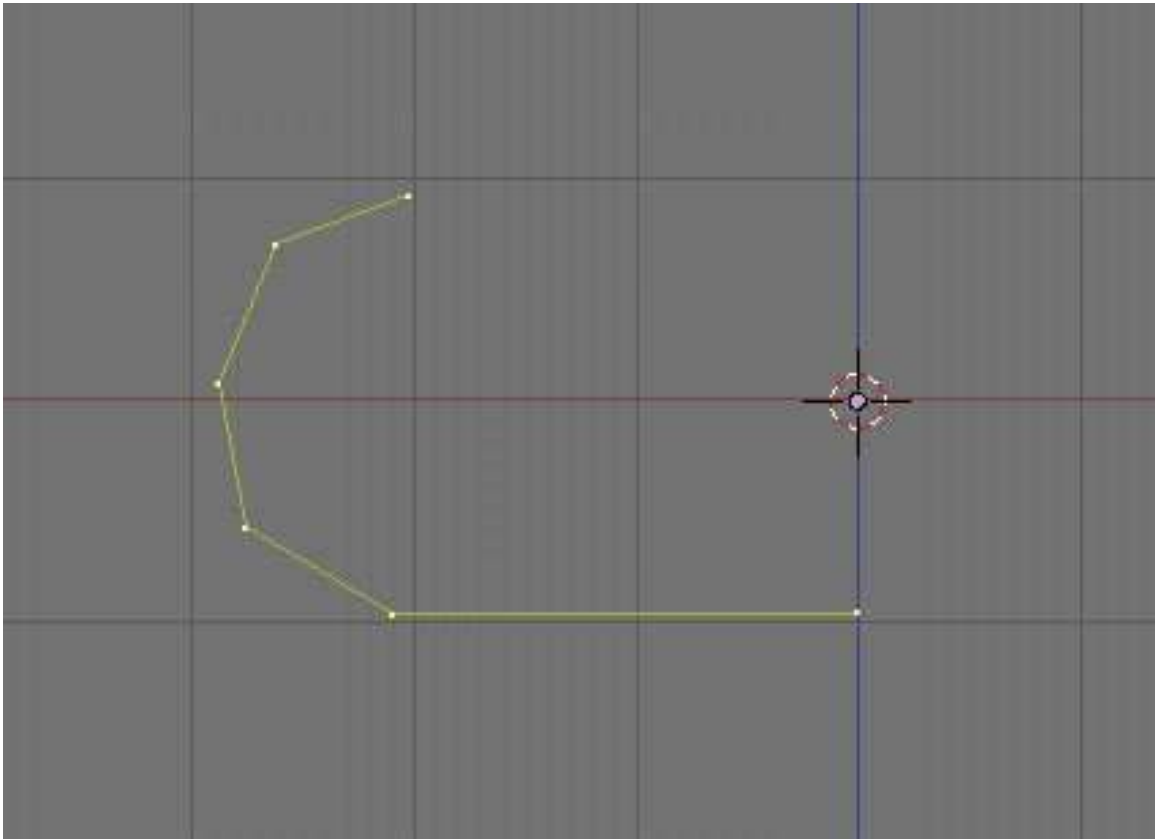


Figure 1403

p Sample Outline Spun with Degree parameter set at 245 degrees

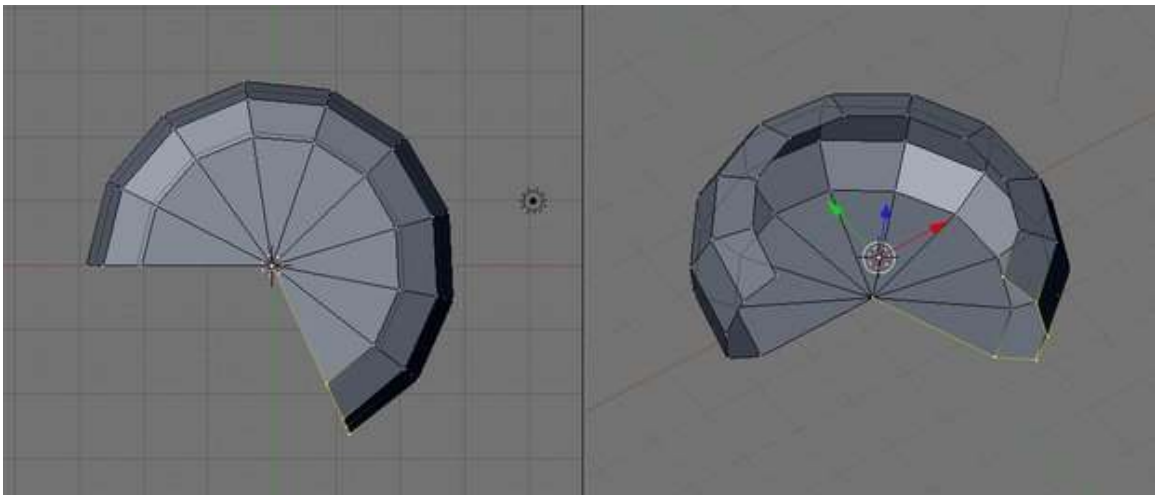


Figure 1404

p Sample Outline Spun with Degree parameter set at 360 degrees

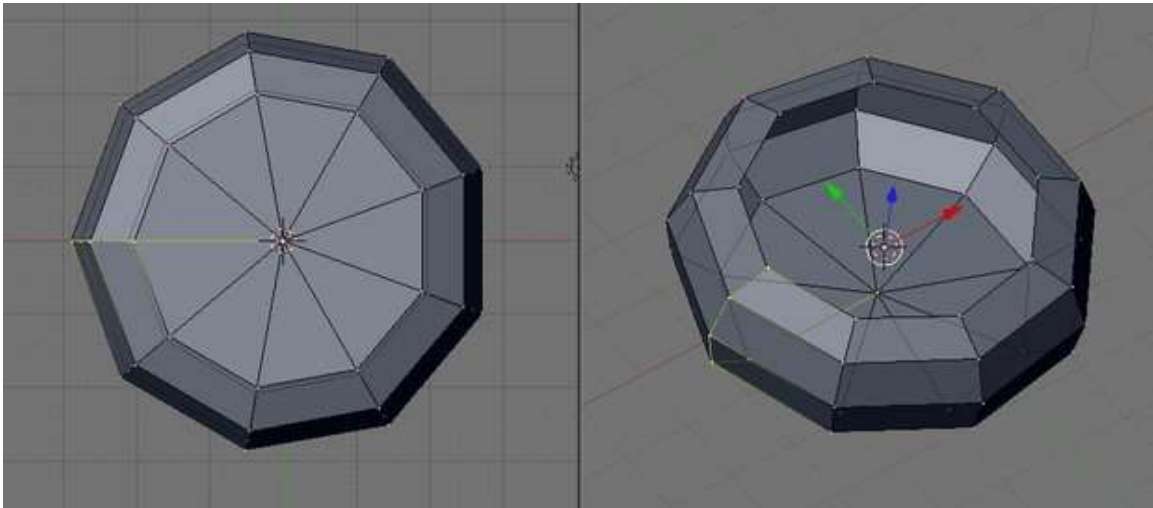


Figure 1405



Figure 1406 none

Steps specifies the number of "copies" of the selected geometry the spin tool makes that it arranges in a radial array around the 3d cursor. A higher value result in much rounder form.

p Sample Outline Spun with Steps parameter set at 4 steps

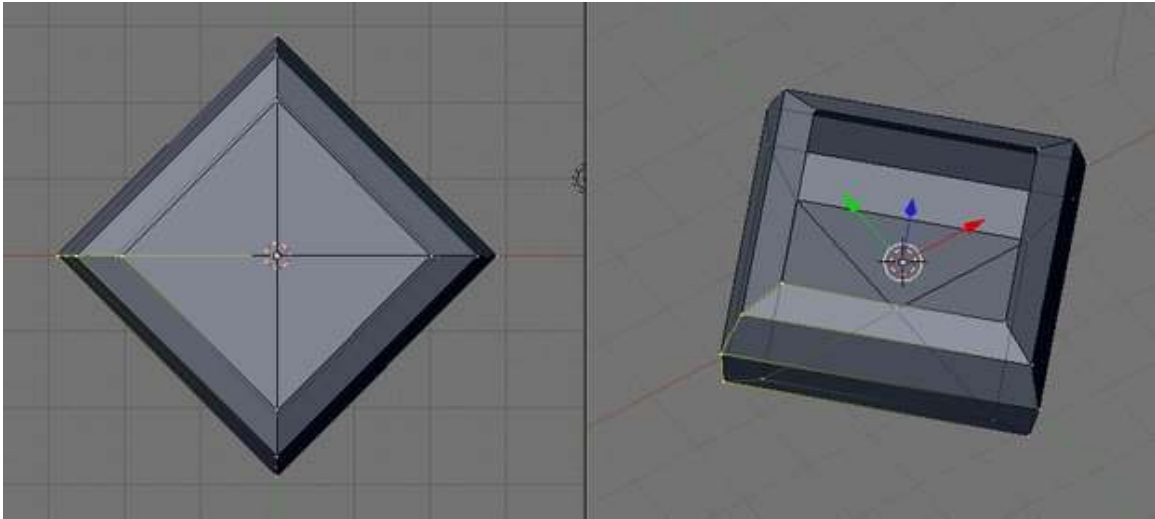


Figure 1407

p Sample Outline Spun with Steps parameter set at 32 steps

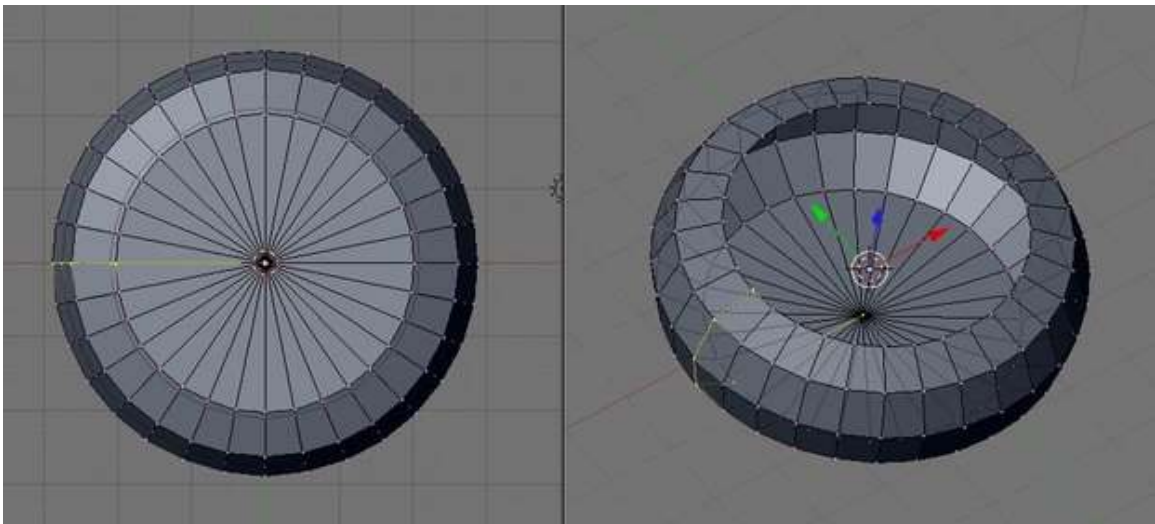


Figure 1408

Cleaning Up the resulting Mesh

After using the spin tool, a little up work might be necessary, especially if you specified a 360 degree turn. Though the mesh looks clean to the eye, the tool actually make geometries (vertexes) that are in the same blender position as another one. So what might look like one vertex may actually be two. A simple way to clean up our mesh would be to select the whole geometry then activating the "remove doubles" from the specials menu (Wkey(Specials Menu) -> remove doubles or Wkey->6key). This

would remove these problem geometries, finds vertices that are too close to each other and merging them to one.

213.3. Post Modeling

After the lathe able parts are done it would be ready to adding non latheable parts. For example a cup may have a bowl-like body, that we could use the spintool. Alas the handle can't be so we had to put it there after. Depending on the geometry added. This geometries could be formed from the mesh itself or be adding in a separate mesh.

214. Illustrative Example: Model a Wine Glass

Lets start with a classic spin tool model. Here we go step by step in making a wine glass... or any other latheable models... using the spin tool.

214.1. Model half the outline

Using what you have learned earlier, create a simple outline of the wine glass. Just the half part and leave the center part open. As a recap, go first to front view. Select the cube (or if you have no object create one) and press Tab to go to edit mode. Delete all vertices of the object, then using Ctrl+ LMB, add connected vertices. Keep holding Ctrl and clicking until finished. Adjust vertices as required.

p Finished wineglass outline

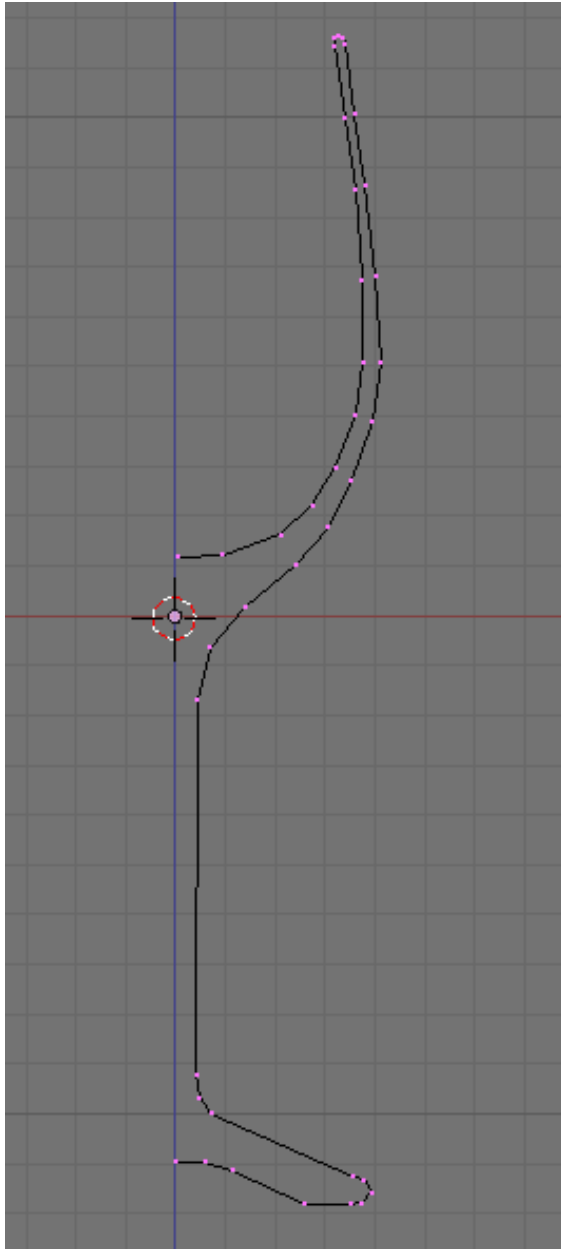


Figure 1409

214.2. Spinning the outline

Remember to align the two end points in the open side. Select the two vertices and scale at the x-axis to zero (Skey->Xkey->0key). Move the 3d cursor to the center by snapping to the selection (Shift+S-> 4key). Then select all vertices by hitting Akey twice.

p Wineglass outline in preparation for spinning

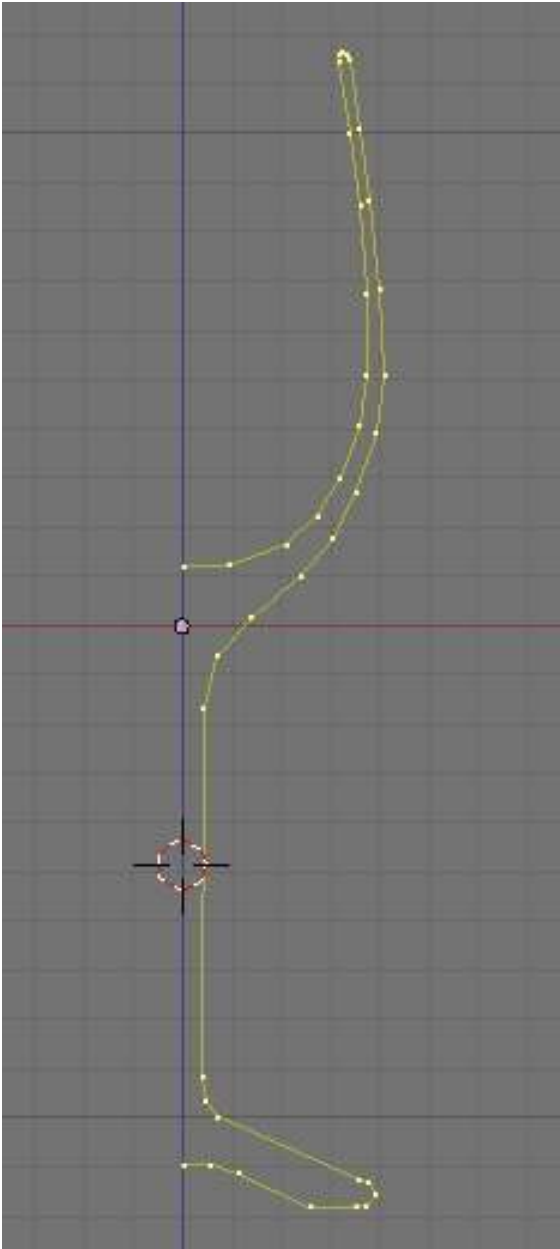


Figure 1410

Now go to top view (Num7). Select the editing button (F9Key) and find the mesh tools tab. Change Degr to 360 for a complete turn and change steps to 20 or higher to form a much rounded shape.

p Parameter settings for the project



Figure 1411

Press the Spin button.

p Wineglass result seen in front view

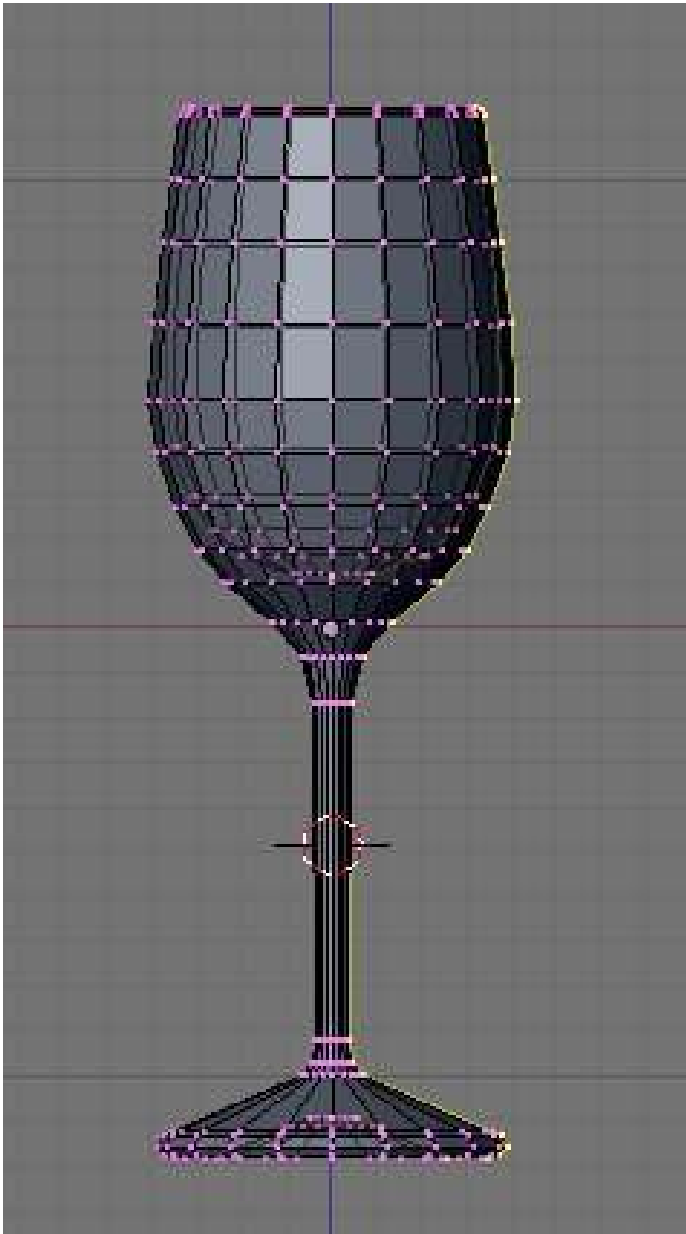


Figure 1412

214.3. Cleaning up the Mesh

Select all vertices by pressing Akey twice then apply remove doubles operation (WKey->6Key/Remove doubles). Now the model is finished. Make it shade smooth using set smooth (Wkey -> set smooth). If you like, add a Subsurf modifier to the object to make it geometrically smoother. If dark bands appear, recalculate normals outside (Ctrl+N), to eliminate them.

p Finished wineglass rendered with no textures



Figure 1413

215. Creating Ogg-Theora movies using Blender

Wikimedia Commons requires that movies be uploaded as Ogg-Theora (OGG) files. As of Blender 2.42a, this is not a builtin feature of Blender. To get OGG files from your finished animation isn't difficult, though. However, you'll need additional software.

There are basically two ways to generate OGG files: you can use one of the many fine video editors or you can use special conversion programs. Video editors like LiVES or Cinerella allow you to load your AVI or your rendered frames, manipulate them, and create the OGG file from it. Please refer to the editor's documentation on how to achieve this.

A disadvantage of a video editor is they are huge pieces of software, duplicating functionality that you already used when you created your animation file/s with blender. It's actually not necessary to install a video editor just for converting your animation to OGG Theora format.

215.1. Converting saved frame picture files to Ogg Theora

It's actually possible to convert frame pictures that you saved before to OGG format movies. The `ffmpeg2theora` software package¹, which is available in source or binary for all relevant systems, is capable of batch-processing files into an Ogg format movie. For example, if your frames were saved as PNG (with filenames `filename001.png`, `filename002.png`, etc.), you could convert them to a soundless OGG file with:

```
ffmpeg2theora filename%03d.png -o output.ogv
```

Sound is possible too, as well as being able to set the quality and framerates. Consult the `ffmpeg2theora` documentation for more.

215.2. Converting AVIs to Ogg Theora

`ffmpeg2theora` can also convert AVI movies to OGG. Usage example:

```
ffmpeg2theora --optimize my.avi
```

¹ <http://www.v2v.cc/~j/ffmpeg2theora/>

216. Creating animated GIFs using Blender and Gimp

This tutorial will guide you through how to make a simple animated Gif Using Blender and Gimp. This is useful for creating Avatars for forums etc. This tutorial assumes basic knowledge of blender and Gimp, see the basic animation tutorial¹.

To start off you will need an animation, this usually should be no longer than 25 frames long.

1. Open Blender. and delete the default cube and add a UV sphere, the default settings for the sphere work fine.
2. Now set the camera size in the scene buttons (F10) to 50 by 50 (I am creating an avatar for deviant art where the required size is 50 by 50 pixels, you may change this if you want)
3. set the frames to start at frame 1 and end at frame 20.
4. select the sphere and insert a LOC keyframe at frame 1 and 21, then go to frame 11, move the sphere and insert another LOC keyframe. this will create a looping animation.
5. Set the image type to PNG or JPEG (it doesn't matter) and render the animation.

Now to combine the images into an animated GIF using Gimp.

1. Open the first image with gimp.
2. Now click File-> open as Layer or press "Ctrl-Alt-O". Select the next frame and it will be added as a new layer. Repeat this for all of the images, or select all of the images by pressing "Ctrl-A".
3. If you press Filters->Animation->Playback it should play the animation. It will probably have a low frame rate making it "choppy". this will be fixed in the next step.
4. Change the frame rate to 40 ms(25 frames per second). Choose file->Save as-> "Name".gif, then choose Save as Animation.

Note: Don't know wheter this is an additional feature of Gimp 2.6.8 or not, but i found out following: In Gimp 2.6.8, within the export wizard, you have to choose option: -save as animation- first and next at the option: -single picture where not mentioned- the option: one single picture per layer. -Sorry i use the german version of gimp-

You should end up with something like this:

¹ Chapter 94 on page 815

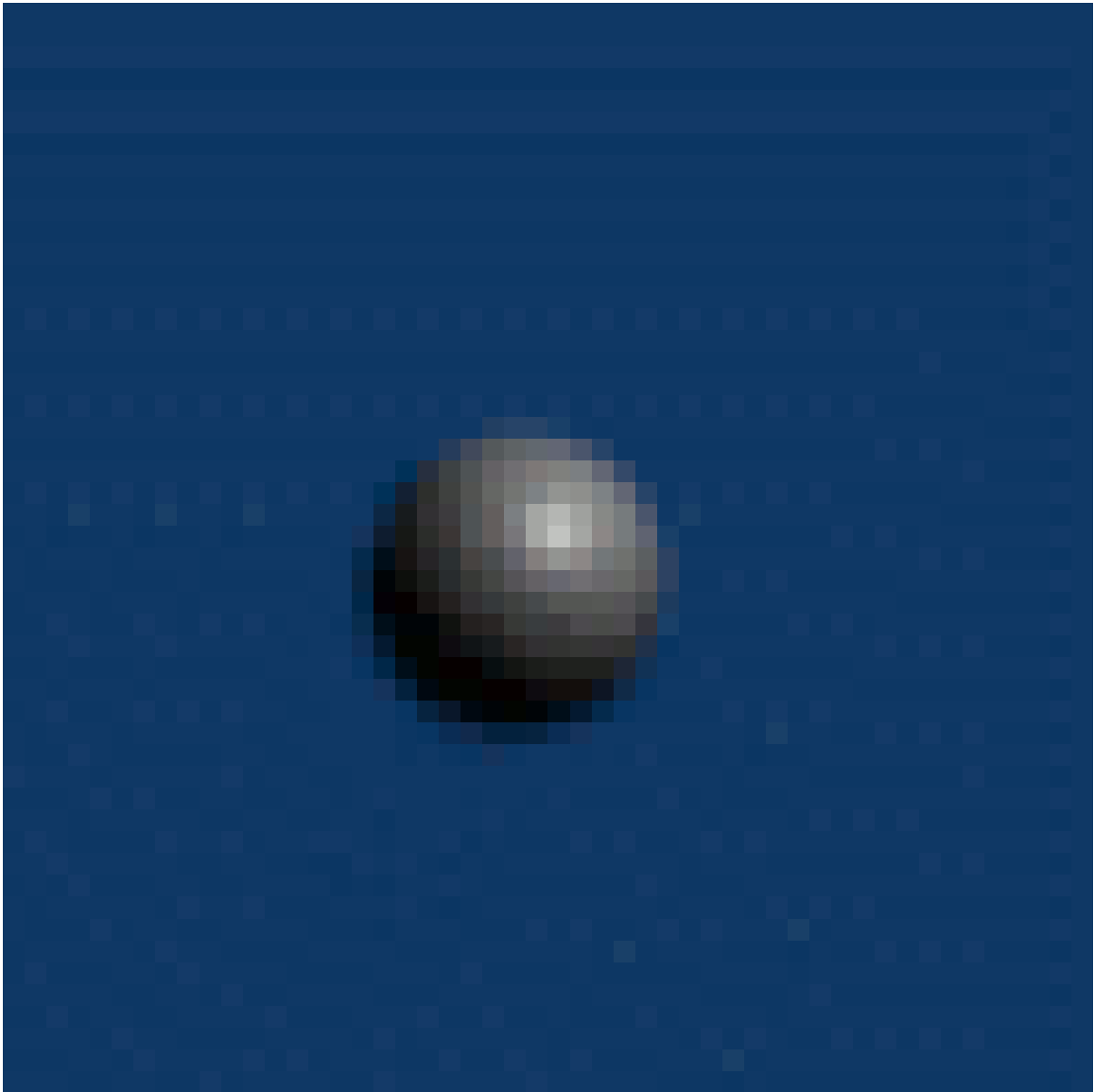


Figure 1414

217. 3D Tiling Backgrounds For The Web

217.1. Overview

This tutorial will guide you through the process of making 3D tiling backgrounds for use with web pages, your desktop, or anything else for that matter. We will be using Blender and a graphic editing program such as Adobe Photoshop or GIMP. I will provide detailed explanation to cater to beginners, however more experienced Blenderists can probably get away with just following the picture diagrams provided. You may also find the diagrams useful if English is not your native language, or if you hate reading instructions.

217.2. Create The Object You Wish To Tile

Start up Blender and look at your lonely cube. We will use this cube as a starting point for the sake of demonstration, but feel free to use any shape you like. Press "**Tab**" to enter **Edit Mode**. Press the "**E**" key on your keyboard and the **Extrude** menu will appear. Select "**Individual Faces**". As you then drag your mouse cursor you will see some numbers move in the bottom left in the window and text saying "Shrink/Fatten". I recommend setting it to 1.0000 to keep it simple. Hold down the "**Ctrl**" key while dragging to do so in set increments.

Next, with the six faces of the cube still selected, press the "**S**" key to scale those faces. I recommend scaling to 0.8000.

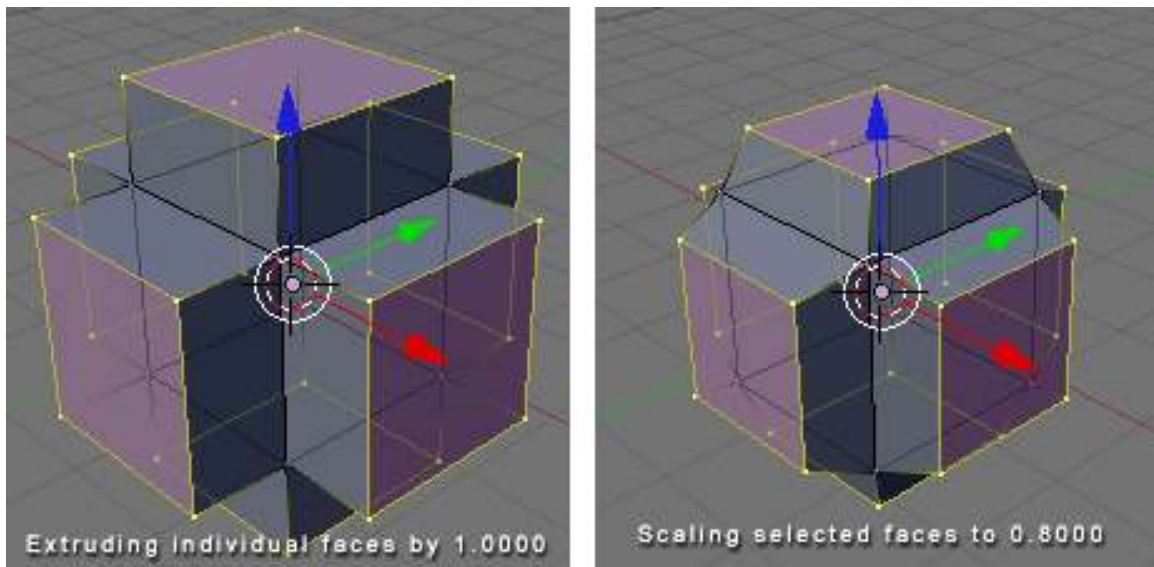


Figure 1415

Good. Now that we have created our object we will prepare to tile it.

217.3. Specify Your Tiling Area

"**Tab**" back into **Object Mode** and press "**NumPad 7**" to go to the Top View, or manually click "**View**" at the bottom of the window and select "**Top**".

Press "**spacebar**" and from the menu that appears select "**Add**" > "**Mesh**" > "**Plane**". The plane has appeared, but is obstructed by our cube object, so press "**S**" to scale the plane to 5.0000.

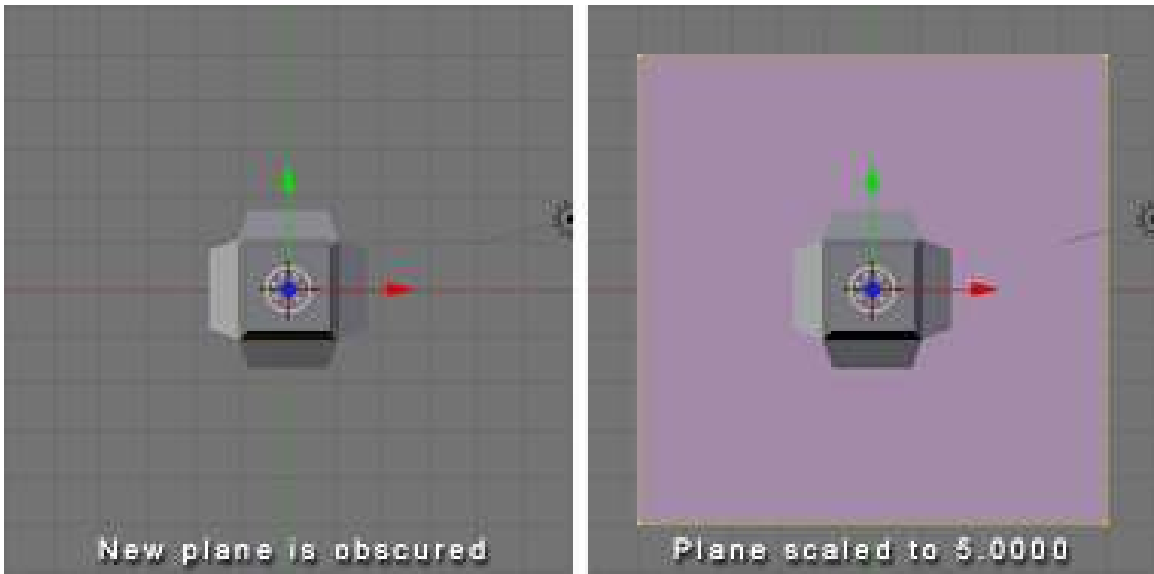


Figure 1416

You'll notice that I've colored my plane black. This is only to make it easier for you to see. You need not bother with this as we will be deleting the face of this plane shortly.

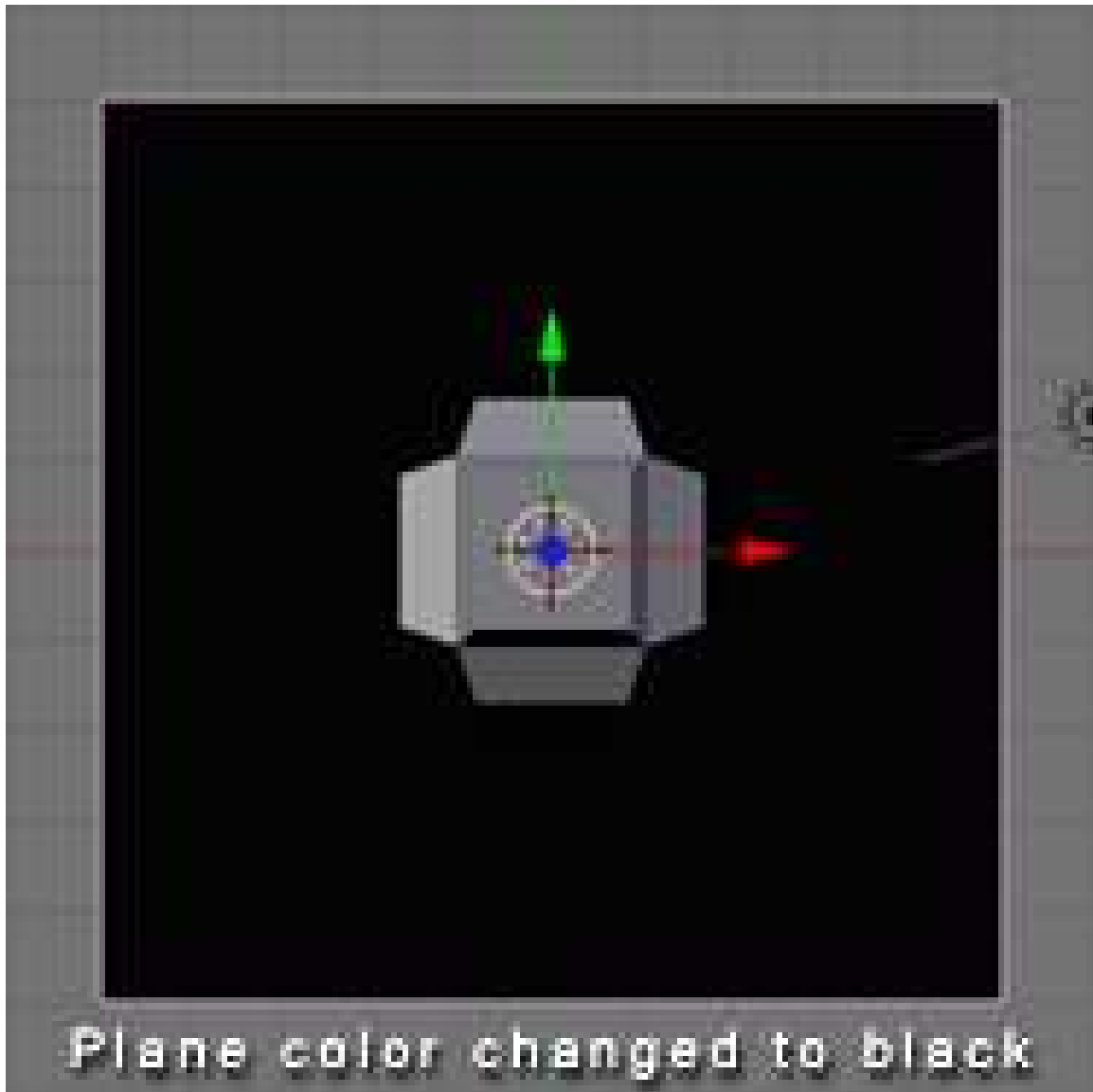


Figure 1417

We now have an easily visible boundary representing the area which will be tiled. If at any point we decide we no longer need or want this we can move it to another layer by pressing "M" and selecting a layer to move it to.

217.4. Tile Away!

Now the fun part begins. If you have created 2D tiling backgrounds before, this concept will be very familiar to you.

While in **Object Mode**, select your cube object. At this point you have a couple of options. One is to make duplicates of the cube object. Another is to make *linked* duplicates of the cube object. I recommend making linked duplicates because you can later edit the mesh of one object in order

to change the mesh of all the duplicated objects. This can be especially useful if you later plan to animate your tiling background (*careful not to distract from the foreground*). It is also useful if you want to experiment with new designs easily without having to re-place all your objects.

With your cube object still selected, press "**Alt+D**" to make a linked duplicate. (*Non-linked duplication is done with "Shift+D"*) Now hold down "**Ctrl**" to move in set increments and move the object to one of the corners of our plane. Make sure the center of the object (represented by a pink dot) is aligned with the very corner of the plane. At this point you may notice it's somewhat difficult to tell whether the center of the object is exactly on the corner of the plane, and that's why we are going to press "**NumPad 5**" or click "**View**" and select "**Orthographic**". For all practical purposes we can spend the rest of the time we are building the tiling pattern in Orthographic View because this view is essential for tiling.

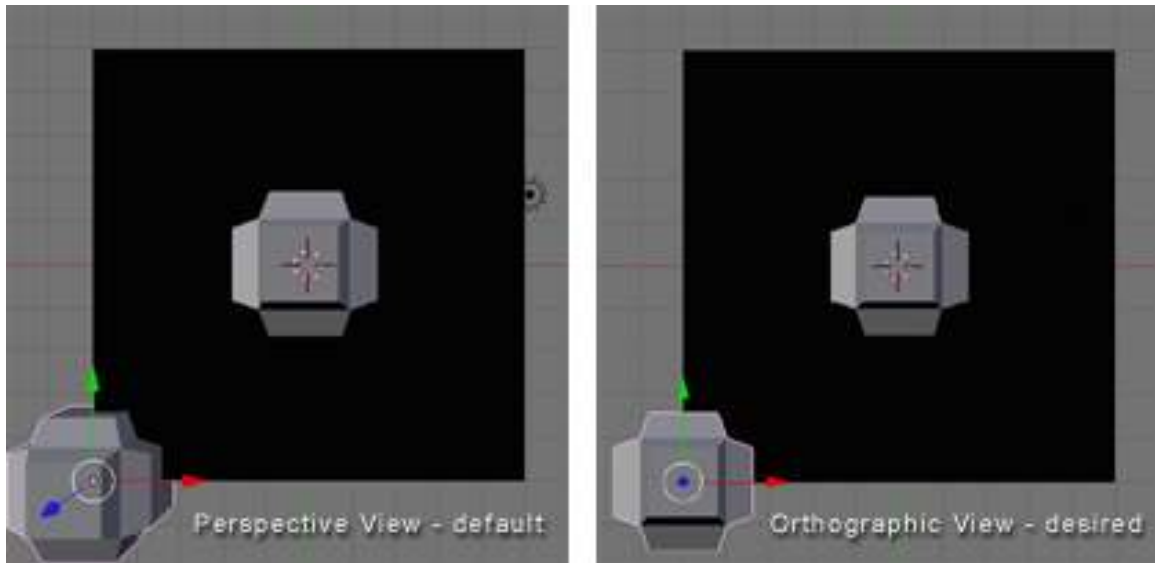


Figure 1418

Now with your duplicate object still selected, scale it ("**S**" key) so that is somewhat smaller than the first. Then rotate it using the "**R**" key. While rotating you can constrain to a particular axis by typing "**X**", "**Y**", or "**Z**" respectively.

We will now make linked duplicates (**Alt+D**) of our rotated and scaled cube object and place one in each corner exactly on the grid (remember to hold "**Ctrl**" while dragging). If you place an object in one corner, you must place it in all four corners because corners touch both the X axis and the Y axis.

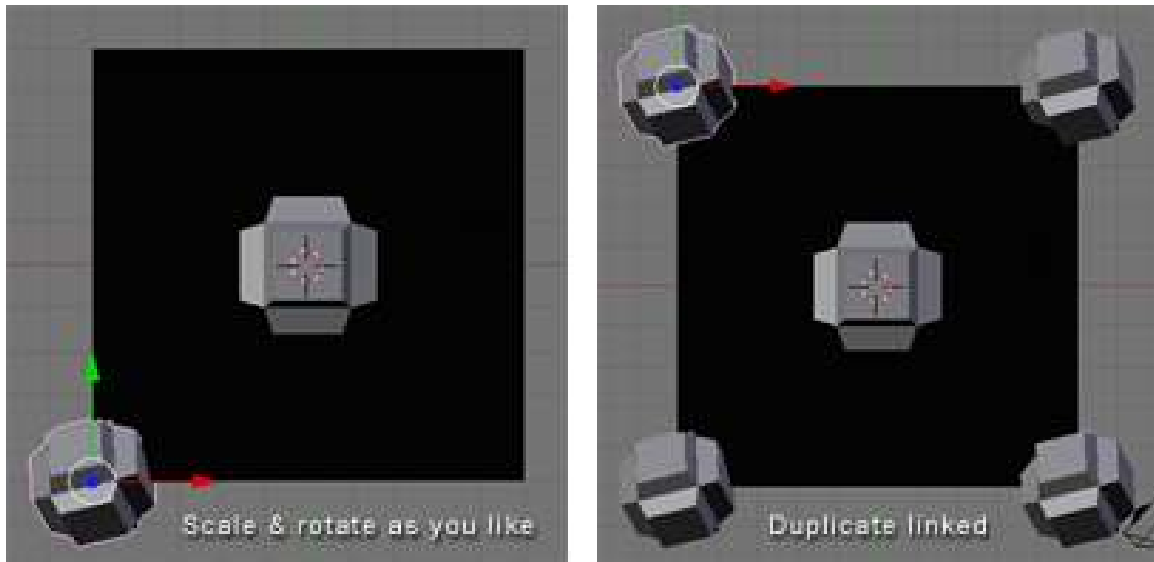


Figure 1419

Now as you will notice in the following diagram, I made an object that is crossing over the edge of our plane. This is good. This is how to make natural tiling backgrounds. However! Whenever we do this we must be absolutely certain to do the same on the opposite side, or else our backgrounds will not tile properly.

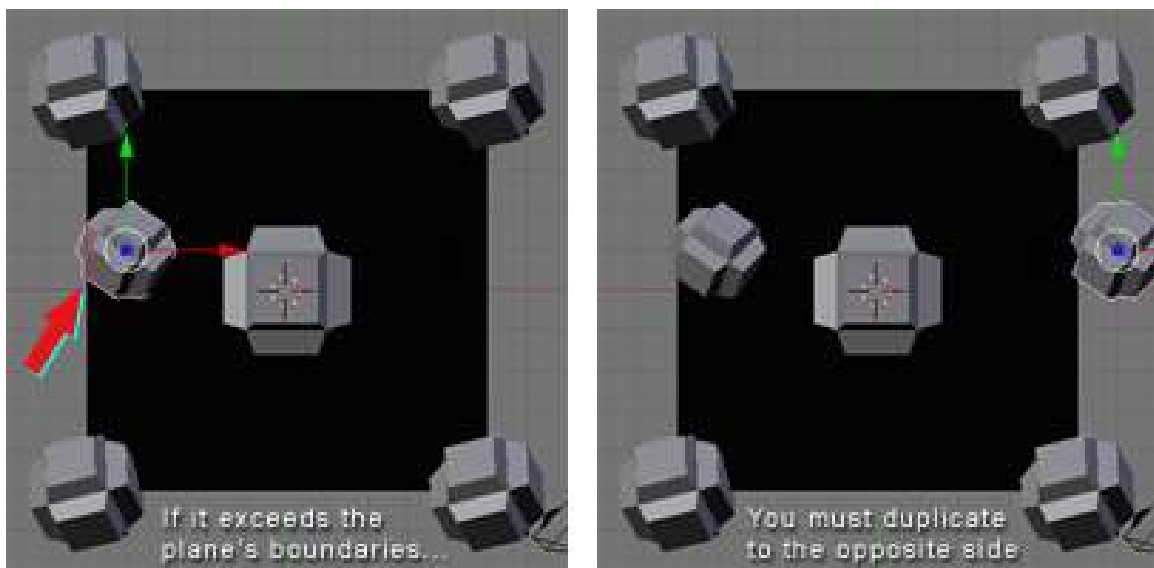


Figure 1420

So now we've created a few objects of different sizes and orientations, however they have all been along the same spot on the **Z** axis. So let's give our pattern some depth - after all, that is the joy of working in 3D!

Before we do this, we'll want to get our guide plane out of the way. So **right click** on the plane to select it, then "**Tab**" into **Edit Mode**. Press "**X**" and from the **Erase** menu choose "**Only Faces**".

Now "**Tab**" back into **Object Mode**. There! We now have only the segments of the plane as our guide.

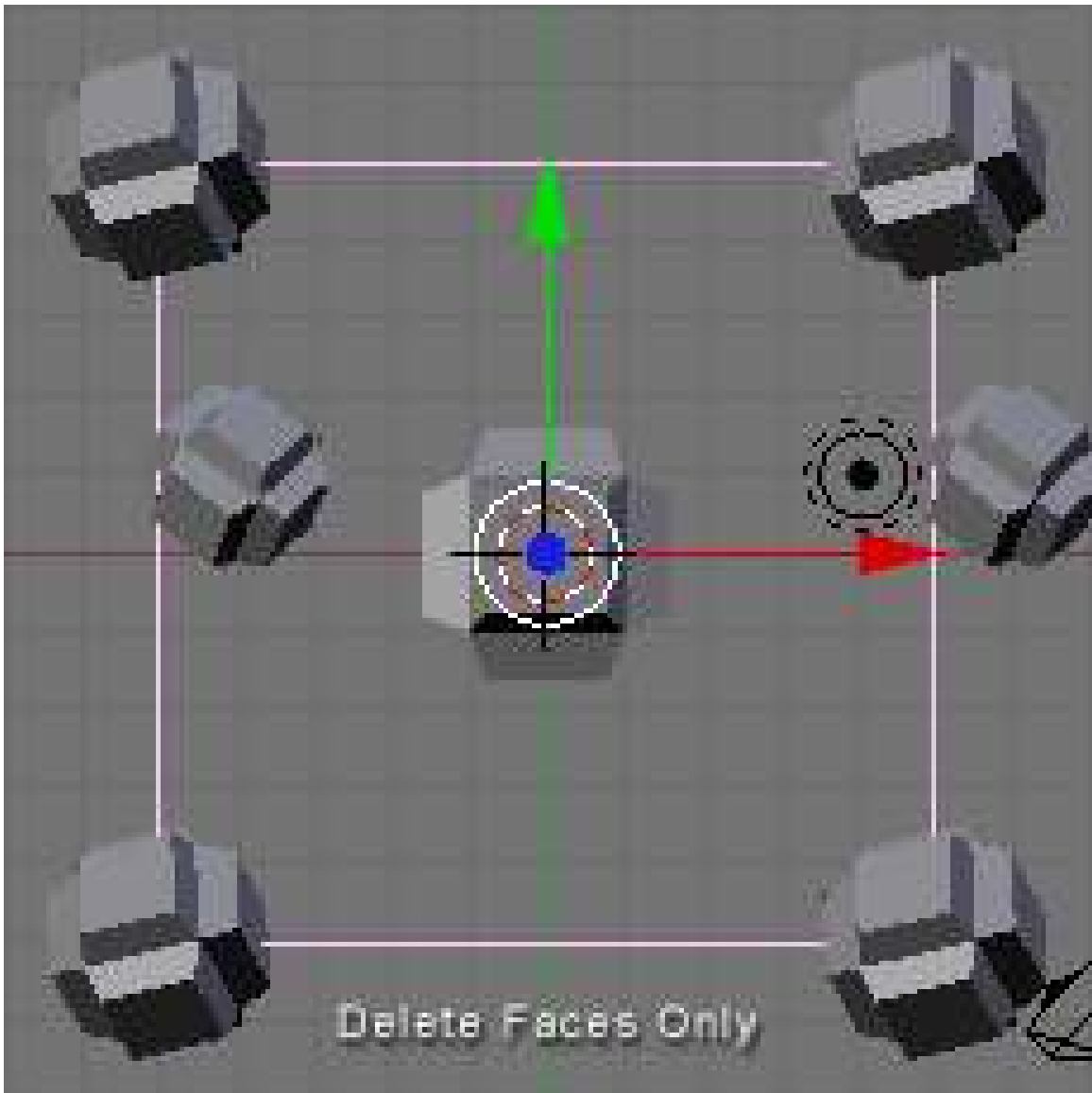


Figure 1421

Hold down the **middle mouse button** while dragging the mouse to see what our design looks like in three dimensions. Now right click one of the cube objects, duplicate a linked copy (**Alt+D**), and press "**Z**" to constrain movement to the Z axis.

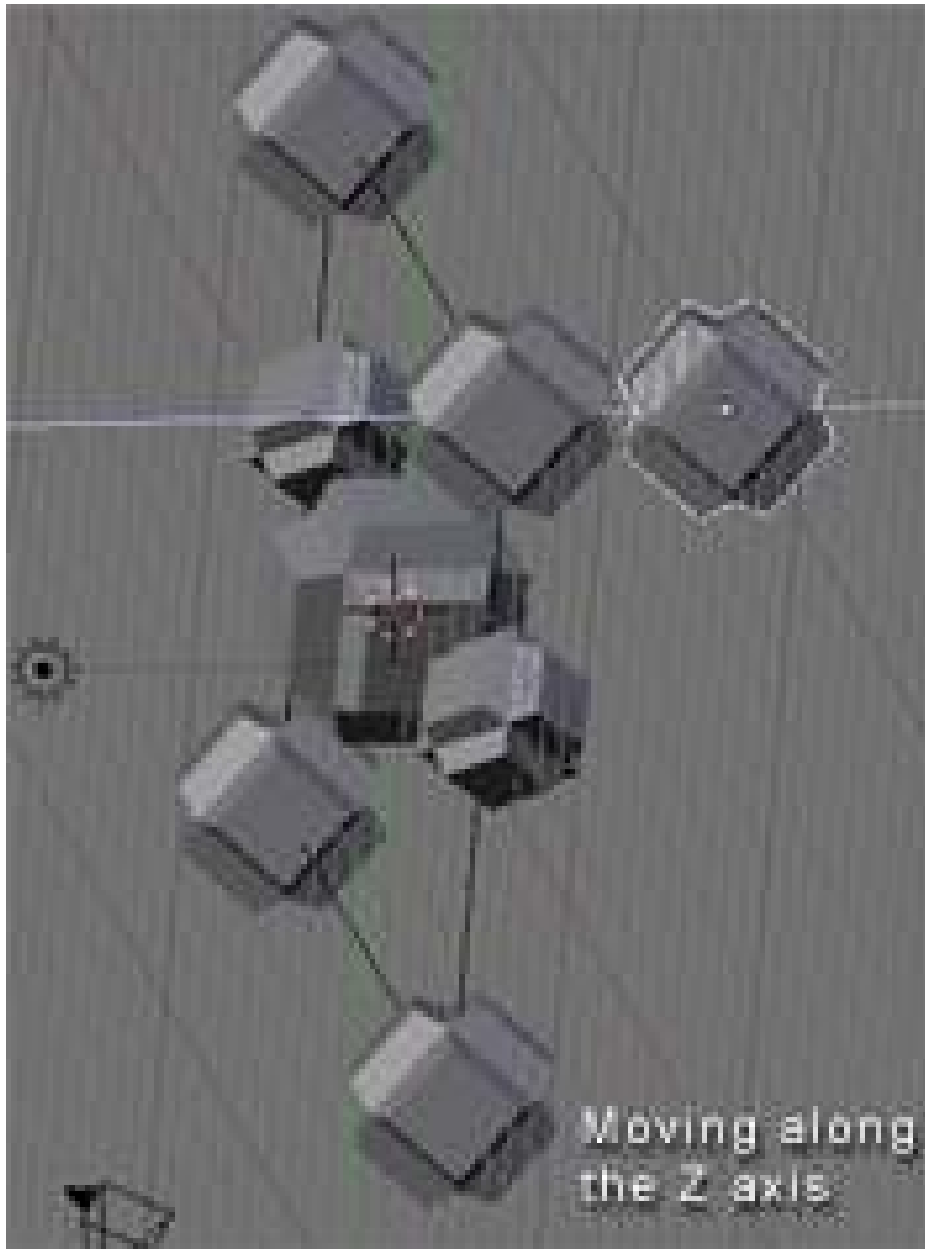


Figure 1422

Navigate in this way along the **X** and **Y** axis as well and once you've found a good spot, **scale** and **rotate**. Remember to make a duplicate on the opposite side whenever you cross the outline of the plane guide. You can go on like this and populate your tiling pattern with as many objects as you wish. The grid is your friend during this process, so you should always have your finger on "**Ctrl**" while moving the duplicate of an object that crosses a border of the guide plane. Also, any object that will "tile" across the border must be at the same point on the **Z** axis as its counterpart on the other side of the seam. Press "**NumPad 7**" periodically to see where you are from a 2D standpoint. This is the view from which we will eventually render the image, so this perspective is the one that counts. Beginners, remember that **mouse wheel** zooms in and out, and "**Shift+middle mouse button**" allows you to "drag" your way around.

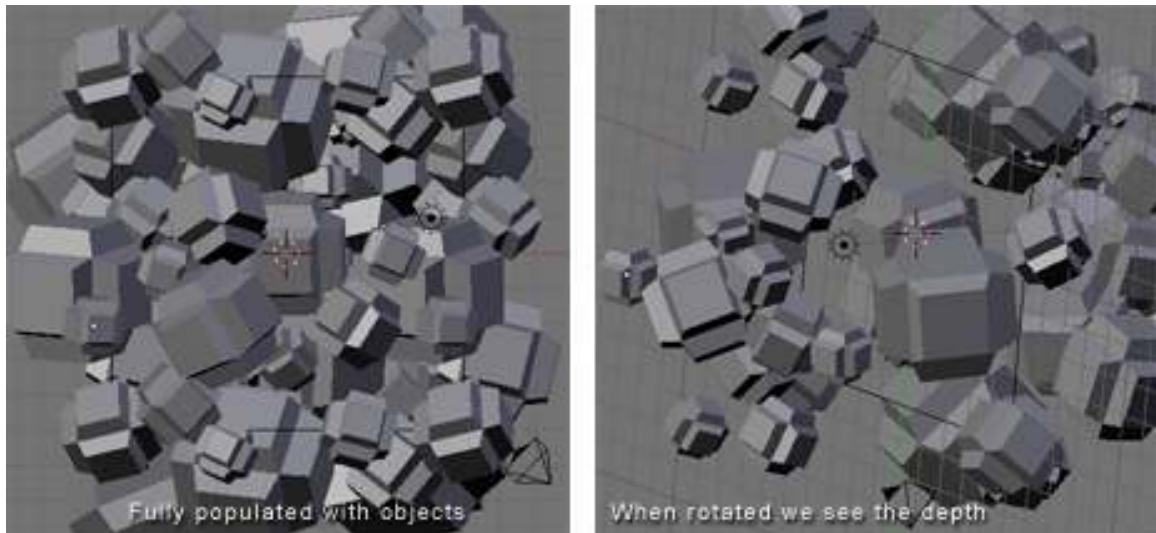


Figure 1423

Keep in mind that in **Orthographic View** we do not perceive the depth of objects. Those far away and those near by appear to be at the same distance. This is what allows us to make a tiled image, but it also limits the apparent depth of the scene. We can compensate for this by scaling down the objects we want to appear further away. Or, as in this example, we can just make each object a different size and place some in front of others.

217.5. Camera Settings

We will now position the camera directly over our cluster of objects. Press "**NumPad 7**", center the view on the cluster using "**Shift+middle mouse button**", and zoom out a couple notches with the **mouse wheel**. From the **3D View Window's** menu, click "**View**" > "**Align View**" > "**Align Active Camera to View**". Now you may be wondering why everything appears to be distorted. This is because when we changed to the camera's perspective, the view automatically reverted to **Perspective View** because by default the camera is set to that view. But just as we changed to **Orthographic View** in the 3D View Window, we can change the view of the camera as well. While in **Object Mode**, swivel the view until you see the camera. It is represented by a pyramidal wireframe with a black triangle atop the opening. **Right click** to select it. Now press "**F9**" or click the **Editing** button (it's icon is four vertices joined in a square). Now in the **Camera** panel you will see a button labelled "**Orthographic**". Press it. Above the button is a value labelled **Scale**. Set it to around 20. (See figure below)

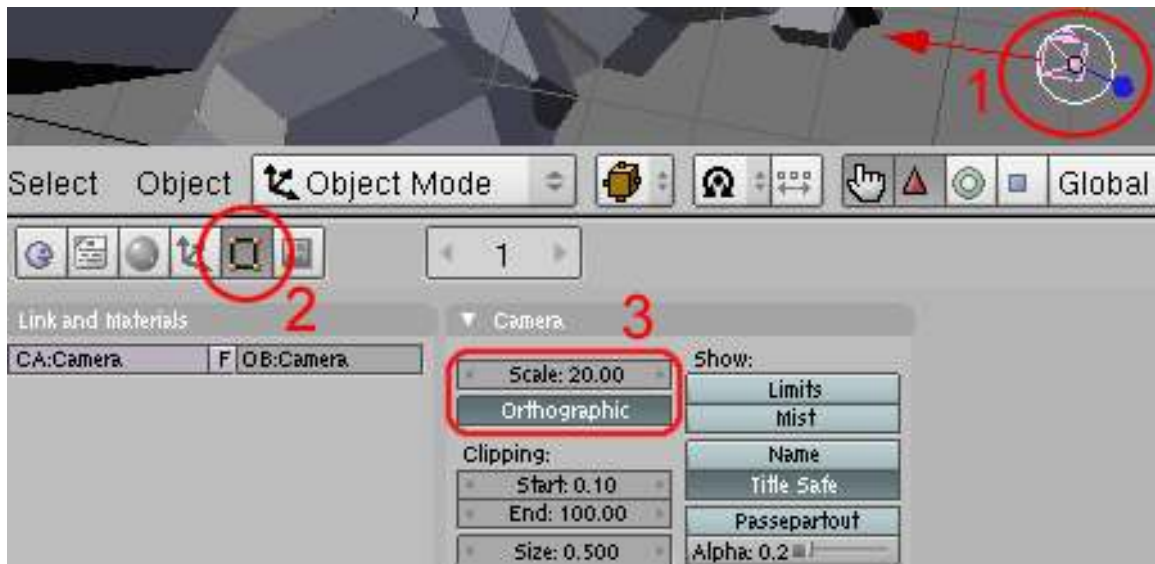


Figure 1424

217.6. Lighting & Materials

Place some lights in your scene, and add some materials to your objects. **Lighting** and **materials** are subjects which demand their own tutorials, so if you don't know how to do these things yet, please consult the Wiki.

Note: I advise lighting your scene pretty evenly in order to make the tiling seem contiguous.

Note: You could also only use Sun lamp, so the light is clean all over the mesh!

217.7. Rendering

Now that you've presumably got your lighting and materials as you want them, it's time to render the scene. Under **User Preferences** click "**Render**" > "**Render Settings**" or just press "**F10**". In the **Format** tab you can choose the dimensions of your rendered image, the file format you prefer, and the quality. I'm going with 800x600 at 100% quality.

217.8. 'Shopping

And it's time to begin post-production. Open your rendered image in your preferred image editor. I will be using Photoshop in this example. Now find and mark the four corners of your tile (on a new layer of course). Just eyeball it.

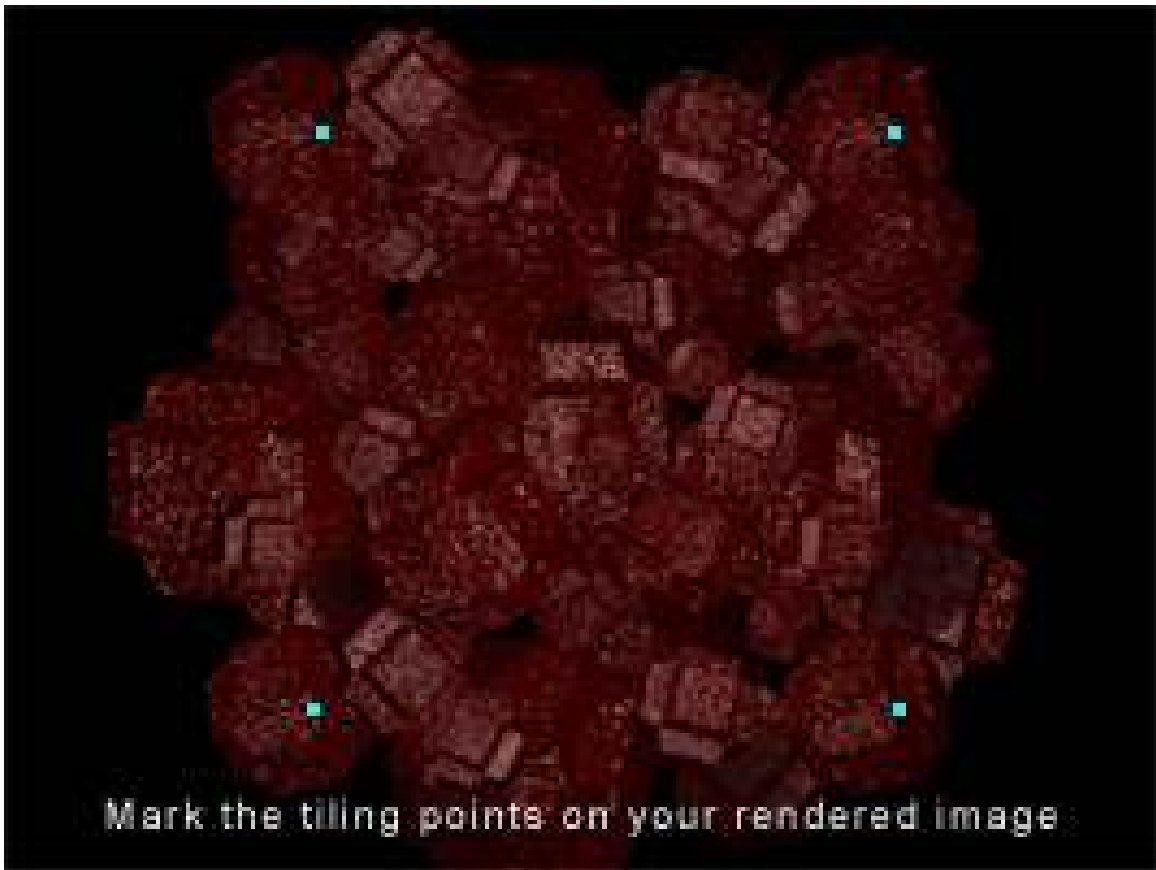


Figure 1425

We will use guides to mark the tile boundaries. In Photoshop guides can be inserted by selecting "**View**" > "**New Guide...**". Guides can be positioned with the **Move Tool**. You can zoom in and make sure your guides are accurately positioned by holding "**Ctrl**" while typing "+" or "-". You can now hide or delete the layer with the markers, as that was only to help us get accurate guides.

Should you decide to make an animated tiling background I recommend layering all your frames on top of one another in your image editor before editing in order to keep the position uniform. You can then animate them using Adobe ImageReady (comes with Photoshop) or find decent low-budget and occasionally free GIF animation programs online.

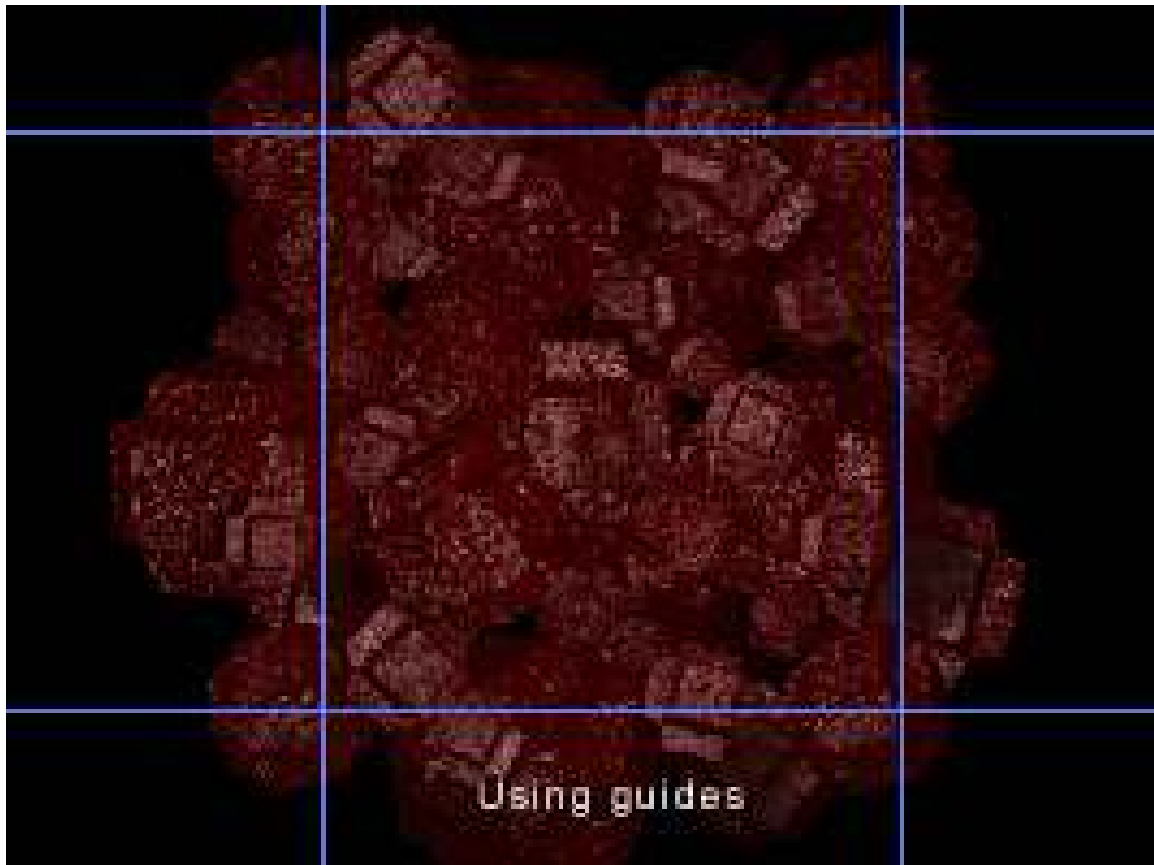


Figure 1426

Now select an area just *outside* of the guides with the **Marquee Tool**. Make sure "**Feather**" is set to **0px** and that the marquee is rectangular and **Style** is set to **Normal** (meaning no Fixed Aspect Ratio or Fixed Size). Also, it will help you to have **Snap** enabled, so go to "**View**" and make sure it is enabled for guides.

Once you have selected the area, copy (**Ctrl+C**) and paste (**Ctrl+V**). Then drag to the right using the **Move Tool** and that piece should snap to the inside of the rightmost vertical guide.

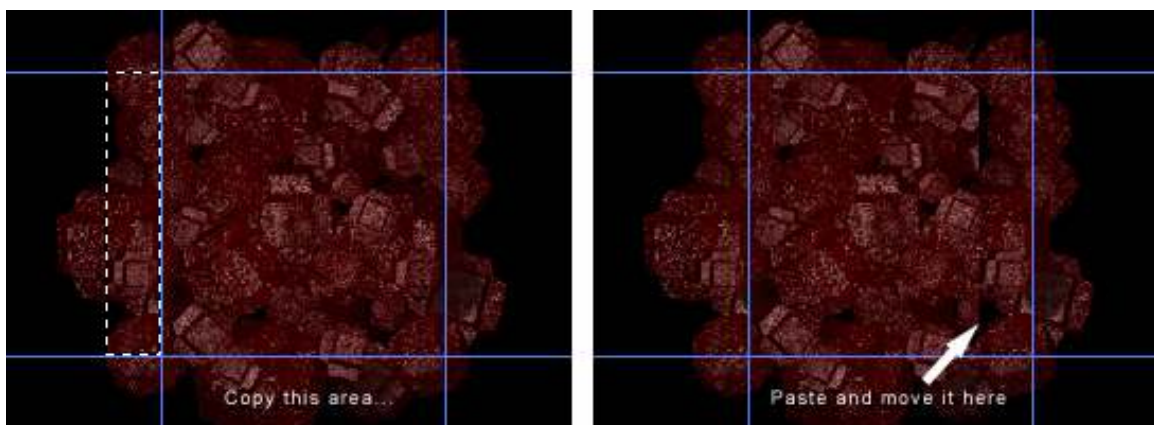


Figure 1427

Select the Eraser Tool and choose a brush size. I'm going with a diameter of 65. You'll want to use a soft gradient-like brush for this. Now erase the left edge of the newly pasted selection so that it blends into the picture. Don't erase any of the right edge.

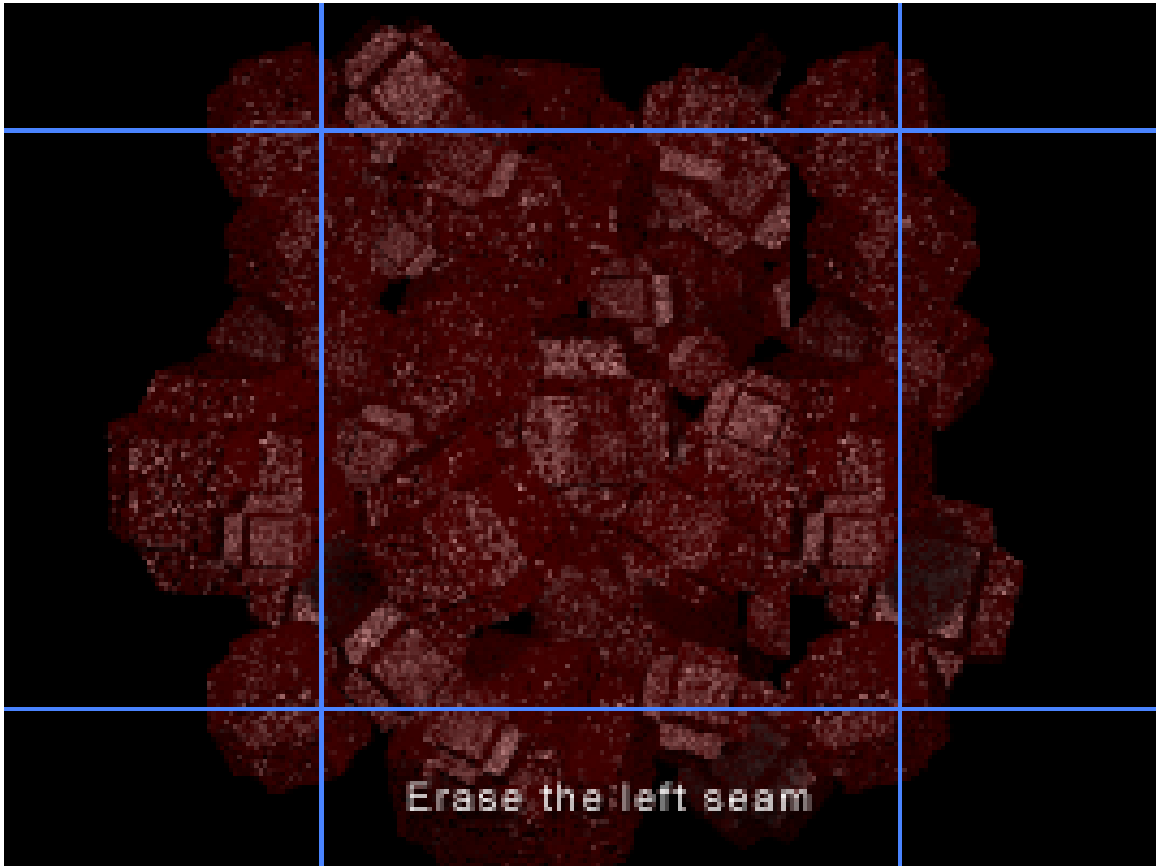


Figure 1428

Press "**Ctrl+E**" or alternatively, select "**Layer**" > "**Merge Down**".

Alright, so that procedure you just performed copying from left to right - now do it bottom to top. Once again, erase the edge. Do a **Save As**.

Finally, select the square center area that will become your tile and do "**Image**" > "**Crop**".

Now scale your new tiling background to a web-friendly size. **Save As** a JPEG, GIF, or PNG. **You're done!**

Note: If your pattern still isn't tiling quite perfectly, load the version with the guides again and repeat the copy-paste-erase process. Remember to merge down your layers each time or the corners may not tile correctly. If still no joy, try tiling the background in a web browser, taking a screenshot, pasting that into your graphics editor, and touching up from there.

Here is the final result:



Figure 1429

218. Cool Things

Cool things in blender that aren't that obvious. Useful tips and tricks in Blender.

218.1. Attribution

Many of The Following Tips and Tricks have been contributed by members of the CGsociety.org¹, a Public Society for Digital Artists. The tips have been extracted from a CGsociety thread². They have been roughly edited to improve readability.

218.2. File Browser Functions

218.2.1. Delete, Move, Rename, and Make Directory

When you are inside a file browser for loading or saving something and you want to create a new directory, just add the name to the path on top of the window and confirm 'Makedir'.

you can also delete(x), move(m), or rename(n) a file. you can do action on multiple files by seling with right click

218.2.2. Preview images when loading them as a texture

Whenever you are loading images as a texture, you can hold down **Ctrl** while clicking on the 'load'

218.2.3. Trick for Creating Quick File Revisions

After you have saved a blend file or an image you can then save it in progression that is:

car .blend. or car .JPG

Next time do "Save As" then press the + (plus) key which will advance the blend file by 1 every time it's pressed. example: car .blend becomes car 1.blend. press again car 2.blend and so on.

The - (minus) key will subtract one. I've gotten into the habit of saving frequently. Yeah I know there is now the undo feature but I like this better because it gives you a history in case you need to back a few levels of a build. You get a saved version at the level you choose.

1 <http://www.cgsociety.org/>

2 <http://forums.cgsociety.org/showthread.php?t=52560&page=1&pp=15>

Note: Blender automatically detects the number... meaning it does not need to be in any position. For instance: If you have a file named 001starport.png or .blend or whatever, pressing the + (plus) key will automatically name it 002starport.png. If you want to name it starport1.png, it will change it to starport2.png.

Two rules: The filename has to have a number. It can be 0, or 1, or 3.141569.

If the file number is a negative, pressing + (plus) key will increase the "magnitude" of the negative number. I may have used magnitude wrong, if so, I mean pressing the + (plus) key will make -0.04 drop to -0.05. The - (minus) key will bring you only to 0, and then it will start to eat itself up.

218.2.4. Open Recent

Control O

218.3. Object/Vertex Manipulation

218.3.1. Constrain Movement to one Axis or to a plane

when moving objects/vertices or set of objects/vertices ([G] key) if you move in the direction of the global X axis (Up/Down) and then press the **MMB**, movement will be constrained to only move in the X axis, or if you move the vertices in the direction of the Y axis and then press **MMB** it will be constrained to move only in the Y axis. The same is true of the Z axis.

You can achieve the same effect by using the X, Y or Z keys while in grab mode. You simply have to press X key, Y or Z, once to lock to X Y or Z global axis,*

To Constrain the Movement to two axes (a plane): Press **Ctrl+X** to move in the Z-Y-Plane. **Ctrl+Z=XY**, **Ctrl+Y=XZ**.*

Alternatively, Select scaling mode and select the axis not to scale with the selecting button at the same time as you hit **Shift**.

- in every case You can hit the X,Y or Z button again to constrain movement to a different set of axes. Normally this different set of axes is local. However you can change the identity of this set to global, local, normal, or view by pressing **Alt+Space**. which cycles through the different identities.

Note: All of these shortcuts work with scaling and Rotating as well

218.3.2. Shrink/fatten mesh in direction of vertex normals

When you're mesh-editing, **Alt+S** will shrink/fatten the mesh selection in the direction of the vertex normals.

218.3.3. Vertex Parenting

You can parent object to a Mesh, in that case you are parenting to the center of the mesh.. BUT if the mesh is translated somehow (let's say by an armature'[S] pose) the center remains in the same spot, and thus the child object doesn't receive any transformation at all.

To solve this, you can parent the child object to a vertex (or a face) within the mesh, and any transformation that the vertex receives is passed to the child.

There are only 2 options, to parent to any 3 vertex within the mesh or to parent to just one vertex. If you parent to 1 vertex then only location information is passed, with 3 vertex all transformations (rotation, location and size) are passed to the child.

How to do it? Starting out of edit mode select the child(ren), hold **Shift** and select the parent, enter edit mode, select one or three vertex, press **Ctrl+P**. That's it!

218.3.4. Work around to welding Verts

For Edge loop (Verts) position both loops together as close as possible then hit **W** then **4** (not on numkey pad). You can adjust how far the effect of collapse can go in the Edit window (a button on the right labeled Limit: *******, where ***** is a number).

As for individual Verts, e.g. Two vertices welded to become one, select both Verts, scale until they are very close then hit **W** then **4**.

you can also do this with the snap combo

select the vertex you want to weld together **Ctrl+S**, Cur -> Sel **Ctrl+S**, Sel -> Cur [**W**], Remove Doubles

Also, whenever Blender pop up a menu with different options, you can just type in a number to choose one of the options (use the numbers not on the **NumPad**)

218.3.5. Make individual objects the camera and Change them back

If you select certain objects and press **Ctrl+0(zero)** it will make them the camera. I use it all the time to align spotlights.

Select your camera and hit **Ctrl NumPad 0** to make it the active camera again

218.3.6. Ordering Meshes in Vertex Groups

If you are preparing to skin your meshes and you are ready to create the vertex groups, you should pay attention to the order in which you create them, because once they are created there is no way to re-arrange them on the vertex-group list.

That means that if you are a ultra-by-the-book person and you would like the vertex group alphabetically ordered on the list then you must create them in alphabetical order.

.. Or, if you would like them to be ordered according to their function (shoulders, then arms, then forearms, then palms, etc. you must create them in that order.

This may all sound like a stupid thing to care about, but If you have a character with 39 vertex groups you may quickly find that when one of them needs fixing it is a little difficult since they were randomly created.

218.3.7. Position and scale along face normal,

-**Shift V**: Position camera along face normal, **Alt S**: scale selected vertices along face normal

218.3.8. Using Fake Users

pressing **Shift+F4** will turn the window into a "Data Select Window" where you can assign and unassign fake users to almost everything by selecting the name and pressing the F key

Creating a fake user allows you to keep useful data blocks (materials, textures, base meshes) at hand even if they are not linked to an object. You can use it to set a default material that would have the shader you like best.

218.3.9. Align a selection of vertices on a plane

If you want to perfectly align a selection of vertices on a plane, you just have to follow these little steps:

- 1.) Before you are selecting the vertices you want to align, position your 3D cursor in the plane that you want to align to (you could select the 4 vertices of a big plane and hit **Shift+S** / Cursor->Selection for example, but you can position it anywhere)
- 2.) Now select those vertices you want to align
- 3.) Choose "3D Cursor" under "Rotation/Scaling Pivot"
- 4.) Now with the [**S**]-key start scaling mode, hit the key of the axis you want to move the vertices on (X,Y,Z)
- 5.) Holding down the **Ctrl-Key**, you can now move the vertices in one line towards the cursor, until the value for the chosen axis is 0.000. Alternatively, just enter, using the keyboard, the scale factor you want (0 in this case) .
- 6.) Hit LMB. The vertices are perfectly aligned along a plane through the 3D cursor.

This even works very well while in perspective view mode, so you can align on the fly and don't have to switch to front/side/top view all the time.

218.3.10. Welding Vertices

You can weld vertices by selecting them in edit mode and pressing **Alt+M**.

218.3.11. View wireframe of hidden Verts

to view the wireframe of hidden Verts, make sure you are in WIREFRAME MODE and then turn SubSurf on and change the level to 0 If you already knew about this then

218.3.12. Select a true loop

shortcut is **Shift + Alt + right Button** of the mouse, serves to select true loop, in vertices as in edges like in faces.

218.3.13. Selecting one object from a single mesh comprised of multiple objects

If you have more than in Edit Mode, you can place your mouse cursor next to one of the Verts in the desired object, then press the "L" key to select all of the Verts linked to that one. "Alt+L" reselects in the same manner.

218.3.14. Precise Zoom and Select/Deselect

Selecting: If you Hold down the **Ctrl+LMB** (left mouse button) and drag the mouse, it will allow you draw a selection as opposed to using the B button which gives you a square.

Deselecting: To draw an area to deselect, Hold down the **Ctrl+ Shift+ LMB**(left mouse button) and drag the mouse,

Zooming: Hold down the **Ctrl+MMB**. Move your mouse vertically to can get a more controlled zoom versus scrolling the Mouse wheel.

This feature may not be present in 2.37 or earlier versions.

218.3.15. Mouse Gestures

Left click and draw:

- a straight line - moves the selected object.
- a circle - rotates the selected object (note: this must be drawn fairly circle-like).
- a V - scales the selected object.

218.3.16. Selecting Obscured or Hidden Objects

Say you are in front or side view and you want to select an object, but it is obscured or hidden behind other objects. If you press **Alt RMB** over a group of objects, a menu will be displayed in the 3D window allowing you to pick the object you wish to select.

218.3.17. Select or Deselect Multiple Vertices

In Edit mode, when you click the **RMB** near a vertex that vertex (face or edge) will be selected, **RMB** again will reselect. By holding the **Shift** key this will allow you to add each selected vertex (face or edge) in that highlighted group.

Pressing the U key (Undo in Edit mode) will also remove the last selection you made.

Alternatively, you can Press B and then draw a box with **MMB**. Anything caught in the box will be deselected. Also works with BB and the draw selection. Draw with **MMB** and you reselect.

218.3.18. Selecting multiple items

You can use right mouse button to select multiple items to append. Doesn't seem to work with files.

218.3.19. Measuring, length, distance on an object.

Hit "F9" (editing), you should have split (2) windows. One "3d" the other "buttons" go to the Mesh Tools 1 panel and press the Edge Length, Edge Angles and the dimensions will appear on your selections in the 3d view.

218.3.20. Adding Connected Vertices

In Edit Mode, if only one vertex is selected, pressing "E" will add a vertex, on a freely defined place, connected to the selected one. As will holding **CTRL** and left clicking the mouse where you want the new vertex to be positioned.

Note: It must be an Image texture and in wireframe mode to be visible.

(2.37a)

218.3.21. Recalculate Normals

Ctrl + N = Recalculate normals outside (you might have to select faces before doing so) **Shift + Ctrl + N** = Recalculate normals inside

These two hotkeys are useful when you extrude some edges and see a kind of seam in between (due to normals pointing in different directions).

Then, after selecting an edge, **Ctrl + NumPad(+)** selects the face associated with this edge. **Ctrl + NumPad(-)** deselect the face.

218.3.22. Create a Quad from two Tris

Alt + J when having two Tris selected makes a Quad.

218.3.23. Remove Doubles

To Remove Doubles use hotKey: W. You can adjust the "limit" option so that "Remove Doubles" has more or less tolerance. This option is located in editing window under the mesh tools panel, (i.e. weld vertices that are further?)

218.3.24. Combine edit levels on a mesh.

When in Edit mode for a mesh (TAB key) you can choose the level that you wish to edit at. At the bottom of the 3D window, there are four buttons. Vertex, edge, face & back-face cull.

By default the vertex level is selected, if you hold **Shift** and press the edge button, you can use both at once.

218.3.25. Change Select Mode

to change select mode (vertex, edge or face) you can press **Ctrl+Tab**. But this way you can't use the Ctrl+Key to ADD the select modes.. Still could be useful, if you don't have a header for the window you're working in..

218.3.26. Precision Warping

When using the warp tool (**Shift+W**) you may find that there are times when you have trouble perfectly closing a 360-degree loop. This is because Blender will warp based on the total width of the selection, which may not necessarily be what you want.

Getting around this is simple, just select two verts to denote the new endpoints, duplicate them (**Shift+D**), scale the two verts times two, so they are just twice as far apart. Then select what you want to warp plus those two marker verts, and warp 720 degrees.

When done, click a vert on your mesh and type **L** to select everything linked to it. Then type **H** to hide it. Once you've hidden all of your mesh, all that remains are the two marker verts. Do a Select All (**A**) and type **X** to delete them. Now unhide your mesh using **Alt+H** and you have just the warped mesh, extra guide verts are all cleaned up.

218.3.27. Precision Cutting

The Knife Tool can actually be quite precise if you take advantage of the snap feature. Press **K**, choose "**Knife(Exact)**" and then hold the **Ctrl** key while choosing where you wish to cut and you will find that the path to be cut will now snap to nearby vertices.

Keep in mind that the vertices being snapped to don't have to be the ones you are cutting. Say you want to cut the midsection out of a UV sphere and you want the cut to be two grid units in height. While viewing the side of the sphere, add a plane, and scale it to be two grid units tall, and wide enough so that it extends beyond the sphere. Now select the vertices of the sphere (because what we have selected is what gets cut) and when you cut it, snap to the four verts of the plane. Now hit **Enter** to apply your perfect cut.

Using guide geometry such as a plane to cut other geometry "cookie-cutter" style can be extremely useful when accuracy is needed. Remember to align your view before cutting, since your view will determine the angle from which the geometry is cut.

218.4. Working with Meshes

218.4.1. Turning Sub-Surfed Mesh into Normal Mesh

If you have a sub-surfed mesh you can turn THAT sub-surf mesh into a normal mesh. Just select the sub-surfed mesh and press **Alt+C** (Conversion).

218.4.2. How to remove (numb) black spots on a Mesh

If a Sub-Surfed mesh becomes (numb) black on some places, that's because of the normals. Select all and press **Ctrl+n** and then confirm. now it should look prettier!

if the above solution does not work , save your Blend file, Quit Blender then restart. Use **Ctrl O** to open the last file and your mesh will have returned to correct shaded state.

218.4.3. Select all holes in a mesh and fill them

Shift+M* selects all Non-manifold edges/vertices (holes) in a mesh **. Then all you have to do is hit **Ctrl+F** to auto-fill those holes with "beauty fill".

Shift+M is an alternative shortcut for 'Select Non-manifold'. You'll find this in the select menu when in mesh edit mode. The listed short cut there is **Ctrl+Alt+Shift+M**. **Shift M** is obviously a lot more comfortable on the fingers though!

• •

Although 'Non-manifold' usually refers to holes in your mesh, it is not necessarily only holes e.g., an edge with three faces coming out of it is also a non-manifold edge!

218.4.4. Fill in four or fewer vertices

select the vertexes and press **[F]**, this will fill in the empty space around them

to clean up a filled in space, select all the vertexes for the area, and press **[F]**. choose OK to make FCon For example: add a plane in wireframe mode, extrude it several times, select all vertexes, then hit **[F]**.

218.5. Animation

218.5.1. Animation Preview in all windows at the same time

It is well known that **Alt+ a** is for previewing an animation on the 3D window. But that's not all of it. Divide your screen into multiple 3D Windows, each from a different point of view.

Press **Alt + Shift + a**

Enjoy!!!

If you have an Action/Ipo Window and 3D windows open, and you issue the **Alt + Shift + a** command from the Action (or the Ipo) window, it will animate both (the action and the 3D) in sync!! Great for visualization of Ipo's effect on your model.

218.5.2. Choose animation mode, Convert mouse movements to IPO

-T in IPO window: Choose animation mode, i.e., linear, bezier, constant -[R] in IPO window: Record mouse movements, and convert to IPO

218.5.3. Animate procedural textures

To animate procedural textures, press 'i' with the mouse pointer in the materials window, and select the type of Key-Frame you wish to set in the pop up menu. advance a few frames, tweak your materials, and set another key frame.

218.6. UV Mapping, Particles and Texturing

218.6.1. Blender color picker

Blender has a PhotoShop-esque color picker. Simply click on the color preview next to the sliders to use it. Hit enter when you have the color you want.

218.6.2. Saving your face groups selections

Regarding UV mapping and Face Groups Selections there seems to be the general misconception that you can't save your face groups selections on Blender.

Most people already know that from within the Face Select Mode (Potato Mode) you can switch into Edit Mode and whatever selection you do while in Edit Mode is passed back to Face Select Mode when you exit the Edit Mode.

Well, Did you ever wonder why Material Index Groups (that are nothing more than face groups with a common material on them) have those little 'Select' and 'Deselect' buttons there? Sure they come handy for later modification of the material index but that is not all about them.

Do this: Before starting the UV unwrapping job, cut your mesh by creating as many material indexes as you need, you can even assign each one a different color so you can be sure that there is no face orphan. Once you have the mesh all cut and sliced (so to speak) you enter in Face Select Mode, then switch into Edit Mode, select the index containing the faces you want to unwrap, press 'Select', leave Edit Mode and Voile! You now have an entirely useful face group waiting for you to unwrap. No more manual (and imprecise) face selection is needed.

If you later need to change the mapping of those faces, don't fear. Just make sure there isn't any face selected on Potato Mode, do as you did first (enter edit mode, select the index, exit edit mode) and there are your very same faces selected again with the UV mapping you already assigned to them.

Note: Another benefit of have precise face selection groups is that, initially, you don't have to worry about UV coordinates overlapping, since you know have the way to select ONLY the faces you want to. For example, you unwrap all your faces by groups and when you are done you can start thinking about scale and position within the texture map, not like before when you have to solve those things as you go.

218.6.3. Bulk Texture Change

Consider a scenario in which you have a scene with a 100 mesh objects, and 50 of them have one texture and 50 of them have another.

If you want to change the texture of the first 50, but don't want to change each individually, do the following. Add a Plane out of the view of the camera. Add your new texture (Material) to the plane. Then use the "Copy to material Buffer" button in the header of the Material buttons. Select one mesh object of the same sort that you want to change, open Material buttons and Paste from Material Buffer.

All the mesh objects with the same texture will now have the new texture.

Alternatively, If you have a material that you want to apply to a lot of objects at once:

1. Select all the objects you want to apply the material to.
2. Apply a material. (this only applies to the last selected object)
3. press **Ctrl+L** > Materials. (this links the material of the last selected object to all the other objects)

218.6.4. Negative Meta-Balls

Add>Metaball as usual. Exit EDIT mode and Add>Metaball. This time before you exit EDIT mode, hit the Negative button in the EDIT buttons window. Then leave edit mode.

If you move the negative Metaball around, you can see the effect it has on the positive metaball.

Be careful though, as negative metaballs are not displayed in the same manner as positive metaballs, you will only see the Pivot point not a meta-mesh.

This is a little test you can try to see the amazing effects negative metaballs can produce.

Make 1 Metaball, make it big. Place three spheres (UV) inside, make them emit particles, one 100, one 200, one 300 particles. Parent three negative metaballs, one to each sphere, and use dupli-vert on

each sphere. Make the 100 duplicate metaball quite big, the 200 medium and the 300 small. Hit **Alt A** to run animation in 3D window.

*(Click here for this author's negative-metaball thread... oops, it's not there anymore, well, not the AVI anyway!)*³

*(Click here for this author's negative-metaball AVI.)*⁴

218.6.5. Maintain the UV layout when moving/scaling/rotating UV co-ords.

When you have the UV image/editor window open and have loaded an image you want to UV map to a mesh, click on the UV tab in the header bar and turn off 'Snap UV to pixels'.

This will help to maintain the UV layout when moving/scaling/rotating UV co-ords.

218.7. Rendering

218.7.1. Tricks, related to the view ports and the render buffers

First. Switching among screens

So you have your screen made off the 3D window, the buttons window and the info window... but you are doing some fine tuning to the mesh in two places simultaneously, and they both need to zoom in the 3D window. You could scroll or zoom out, translate the view and zoom in again. None of them an elegant solution.

Another situation. You are working on a model and are using an image for reference. You are not tracing over the photo, just take a look at it often to make sure you don't deviate to much from the concept. So you open the photo in a 2D program and keep switching back and forth from Blender.. or you have the photo open in an image window and keep maximizing and minimizing the window... another hassle

Worry no more!!! Blender can handle multiple virtual screen (ala Linux) and you can come and go from them with just one key stroke.

Just press **Ctrl+Left Arrow** or **Ctrl+Right Arrow** (for all of you OS X users, add a **Shift**) and you are switching screens. Go ahead! By default EVERY .blend file comes with 3 screens ... and of course you can add/delete as many as you see fit.

the magic button to add or delete screens is right beside the Tools menu, up there in the info window.

218.7.2. Using the render buffers

Ok, so you set your scene and press RENDER, a nice window comes up and you see your hard work coming to existence (that's the default behavior, if your change it on the display buttons then this may not work for you).

³ <http://www.elysiun.com/forum/viewtopic.php?t=14058&highlight=metaball>

⁴ <http://www.selleri.org/Blender/movies/Metaball.avi>

Do you realize that the window containing your render image is also a render buffer? Actually they are 2 buffers for you to play with. Whenever the render window is open (and you can re-open it by pressing F11 without having to wait again for the render) if you press the J key you can switch from Buffer A and B. (the last active one is what you save when you press F3). You can even switch buffers in the middle of a rendering (but I advise against that when rendering very complex scenes, you have been warned!)

The cool thing about having two separate render buffers is that you can have instant before-and-after images for things that you change in the scene. For example you are searching the perfect position for a light source in a scene, you place it and do a render, place the light in another position, switch to the second buffer and do a new render. Now, with the render window open, just press J to see how the change on the light's position influence your scene and that makes your decision easier.

By the way, the render window can be zoomed (by the normal ways or by pressing Z) to do a closer inspection of the image.

218.7.3. Render window Tricks

To zoom into the render window, use the **ZKEY**.

To find out what the (Red, Green, Blue, Alpha) values of rendered image are, left click and hold the button of your mouse. This will reveal the RGBA values of the pixel below the mouse cursor. i.e. R 127, G 255, B 13, A40. The values will appear in the bottom left corner of the render window. You can also hold the button and drag the mouse around. This will display the values of the pixels your mouse pointer passes over.

With the render window open, you can press **AKEY** to view an alpha version of the image. Press **AKEY** again to go back to the normal colour view.

To do a before/after comparison after making a change, hit **JKEY** to switch render buffers, then hit **F12** to render. In the render window, use **JKEY** to alternate between the previous render and the current one, so you can easily see the differences.

218.7.4. Working while you Render

If you use a X/X11 based window manager, you do not need to watch blender while it renders, go to a different virtual desktop. Blender doesn't have to keep X informed of what's going on and rendering speed may increase.

218.7.5. Border Rendering

In the rendering buttons find the buttons marked "Border" and "Crop". If you depress "Border", you can get a rendering of any part of what the camera sees. Just do the following :

Go into a camera view using **NumPad 0**, press **Shift+B**. Then, mark the limits of the rendering as you want them using LMB. Next, render the usual way and the section you marked will be rendered first, then it will be integrated to the complete rendering. If you wish that **only** the chosen section would be rendered then click on the button marked "Crop" also.

218.7.6. Creating a cluster of particles which takes little time to render

Render some particles, and make the picture into an alpha mapped tga in GIMP or whatever graphics editing application you use.

Load the image onto a plane. Adjust the alpha settings accordingly. Parent the plane to the emitter. Press dupli-verts.

You now have a cluster of particles which takes relatively little time to render. Of course it doesn't have to be an alpha map of particles. That's entirely up to you.

218.7.7. Alpha from render view

When cut & pasting stuff from render window to {insert your favorite image editor here} using **Alt+PrtScr**, cut and paste the render first, come back to render window and press "A". It changes the view to alpha and you get black & white mask to cut the background nicely in the {again, favorite image editor}. Nice when you do testing in low res.

218.7.8. View alpha texture as wire.

Ctrl+d in 3d very usefully for preview without rendering.

Also, if you have an object (works best on a mesh plane) with an image texture, you can use **Alt+V** key outside edit mode. This will adjust the object's size values so that the image won't be stretched when projected.

218.8. Miscellaneous

218.8.1. Restoring your "lost" work.

If you go to the temp folder where you have Blender save its temp .Blend files, reload the most recent one and this will save you losing the whole of your work. You don't even need to save a .Blend file for this to work. You can change the settings for this in the tool window at the top of the Blender screen.

218.8.2. Built in Hot-Key List

Since the few last versions (since 2.28 I think) blender has a built-in all inclusive hot-key list. Just open a text-editor window, and right besides the option to create a new text-buffer there is a menu option called "KEYLIST.intr". Select that option and the full list is loaded in memory

--Edit: There may be many hot-keys missing, especially the newest ones, like M to mirror a mesh, K for the Knife tool, **Alt+Z** for textured view... but the list is a good start at least.

218.8.3. Un-compiled PlugIns

Blender can load plug-ins for texturing, sequence editor, etc. . However, Blender comes with a few of such PlugIns un-compiled.

In Linux they are located on the plug-ins sub-directory of the default Blender install, and all that you need is a Make command to compile them. I don't know how to compile them in Windows, but there they are, just waiting for you to awake them!!!

218.8.4. Blender “Easter Eggs” (Weird things included in blender)

All Publisher versions of Blender have been shipping with a Monkey mesh called Suzanne. Just open the main tool box (**Space Bar**) -> Add -> Mesh -> and right below the other primitives you'll see the Monkey.

Why/What it is for? Only NaN programmers know*. It is supposed to be a private joke among the blender developing team. Since then, its considered as a sort of mascot for blender. However, it is incredibly useful as a quick complex object for testing textures, shaders, etc..

By the Way, Suzanne isn't the only joke included... but I won't spoil the surprise. You will bump with them on your daily work, that is for sure.

- NaN was the company that originally developed Blender

Try this: run blender with the -Y argument (open the command prompt/terminal, go to your blender executable file folder, and type blender -Y)

218.8.5. Weird Error Message

Sometimes, when trying to use a boolean on a tri-based mesh, Blender gives the prompt: "Wanna Crash?" >Yes Please!

but clicking it doesn't crash. This is because Blender is a lot more stable now compared to when that 'crash' request was coded. It also used to appear when using beauty fill after face fill. (**Ctrl+F** in edit mode)

Blender 2.37 now has a 'widget', which replicates the red/blue/green axis symbol in 3DSMax in the 3d windows. Rotation scaling and movement of objects/Verts/faces etc., can be manipulated using the widget, or in the usual manner of earlier versions of Blender.

218.8.6. Turn your blender animation into a screen-saver (Windows)

to turn your blender animation Windows binary file into a Windows screen saver, rename the EXE file into SCR and right click it and install !

218.8.7. Discover the FPS rate of a Window

If you hit **Ctrl+Alt+T** key in a window, Blender will tell you how much time it takes to render a single frame of that window, in milliseconds. Valuable Benchmark

218.9. ???

using construction widget press **Shift** to get precision mode for fine tuning. Release left mouse button (LMB) and holding **Shift** down press it again, then you'll get moving along another axis.

218.10. Sculpt Mode Hotkeys

- F: an interactive brush resize
- Shift F: an interactive brush strength adjuster
- Ctrl F: in interactive texture angle adjuster for your brush.
- Shift B: a rectangular zoom selection for close-up work
- Alt B: hides all but selected rectangle
- A: toggles airbrush
- S: smooth
- D: draw
- G: grab
- L: layer
- I: inflate
- P: pinch
- V: toggles add and subtract in draw mode
- Use X, Y and Z to toggle axis mirroring.

219. Troubleshooting

219.0.1. ATI Radeon Slowdown Problems

Go the ATI site driver downloads section and select the appropriate OS/graphic card. This will (most likely) link you to the d/l for the current Catalyst 4.4 drivers.

At the bottom of that page is a link to "Previous driver versions".

D/L the Catalyst 3.7 driver package. Run the EXE and it will extract the driver install package to:

C:\ATI\SUPPORT\

In this directory will be a directory named (for XP/Win2K users: \wxp-w2k-7-93-030812a1-010735c-efg\

This is where the driver install package is located. DO NOT RUN THE SETUP. You don't need to install the old driver.

Navigate down thru the dir structure to the following dir: C:\ATI\SUPPORT\wxp-w2k-7-93-030812a1-010735c-efg\2KXP_INF\B_10679

In this dir you'll find a file called: atioglxx.dl_ This is a "packed" version of the ATI OGL driver.

The next step specifies how you can extract these files to your hard disk using the 'expand.exe' utility included on the cd.

The 'I386' folder on the Windows XP CD contains a utility 'expand.exe' that can be used to uncompress all compressed dll files. It is a commandline utility, so you will have to run it from either the command prompt or the Run dialog. Some examples of its usage are:

```
expand X:\I386\ADROT.DL_ C:\ADROT.DLL
The above command decompresses the compressed DLL
```

ADROT.DL_ on the WinXP CD, copies it to C:, and changes the extension to .DLL .

Just extract that file (atioglxx.dll) to your Blender install directory. Usually C:\Program Files\Blender Foundation\Blender\

Launch Blender.... no more slowdown.

Addendum: I found this solved a problem of crashing on start up after updating my Catalyst driver from 7.2 to 7.8, that is expanding the old 7.2 atioglxx.dl_ to atioglxx.dll and copying it to the Blender directory.

219.0.2. For people who the above solution leads to Blender crash at start up

Thanks to Xenobius at Blenderartists Forum

This solution is tested on ATI radeon express 200 (x200) integrated video card. However it should work on other ATI card.

Download the Nvidia 53.03 Driver [HERE](#)¹

Extract the driver to C:\NVIDIA

Copy the nvoglnt.dll from the C:\NVIDIA folder to your blender program folder (i.e. C:\Programs Files\Blender 2.40\)

Rename nvoglnt.dll (the one in the blender program folder) to atioglxx.dll

Run blender...voilà! Blender runs like it should!

[This works, because it can't use proper gl calls and uses no hardware acceleration so speed goes crappy...]

¹ http://www.megagames.com/news/redirect.cgi?http://download.nvidia.com/Windows/53.03/53.03_winxp2k_english_whql.exe

220. Creating Blender Libraries

You've finally made that perfect object, armature or material: a gamepiece, a robot, a fully rigged vampire character, or a millimeter-accurate model of the Empire State Building. Besides using this work in your own artistic project, you have made the courageous decision to share that creation with the world.

Sharing your creative work is a great way to "give back to the community," even if you don't write Blender code and you can't translate the Blender documentation to Swahili.

220.1. How to Make a Library

So how do you make a library? In Blender, you don't need to export to a special format. In fact, you don't need to do anything special beyond saving your regular .blend file. Every .blend is a library file. Users can Append what they like from your .blend file, and ignore parts which they don't need for their own project.

This scheme has some benefits and some drawbacks.

The benefit of using .blend files as a library format is that it's super easy to include extra stuff to help the user see the objects. If the user loads the .blend file directly, it works like the pretty packaging for foods, including a quick and easy way to get a pretty "serving suggestion" rendering of the library contents. What you save in your .blend is what the user will see when they load it, including all your user interface settings, lighting types, and camera positions.

The drawback of using .blend files as a library format is that it's super easy to include *unintended* things, such as extra meshes, unused material and texture channels, and other things which the user will not find helpful. Blender doesn't save things which are no longer referenced anywhere, but it cannot read your mind if you leave a spare mesh on layer 13 which uses some abandoned Material.034.

Also, some people are not accustomed to the way that Blender saves all of the user interface settings along with the .blend file. When they load your mesh, they see your way of working. This can be instructive, but unless that's your intention, it's best to try to stick to a simple and clean user interface setup for your library files.

For best results, you need to apply some discipline to publish a clean and useful library.

220.2. Library Usefulness Checklist

After having used several library .blend files from various sources, I propose that anyone making libraries follow a few suggestions:

- keep to a certain object, theme or area of focus for each library file
- all the test cameras and lights prefixed with a dash; e.g., **-Camera**, **-HemiLight.001**, etc.
- any other components not intended for Appending prefixed with a dash; e.g., **-ground**
- all the test cameras and lights on the last one or two layers (**lower right layer button**)
- any composite object intended for Appending organized in a **group** to hold it together
- any groups, objects, materials, textures intended for Appending given **meaningful names**
- document your unit scheme; e.g., **1 blender unit == 1 imperial foot**, etc.
- any other layer contains test-render-ready objects or scenes
- choose rendering and world **settings which will not take an hour** to make a simple test render
- make visible upon loading the layers required for a camera, a good lighting angle, and a shared object
- make visible upon loading one small text file which lists layers and objects
- make visible upon loading any python script, **with instructions on how to start it** in a big comment
- make your licensing expectations clear: **artistic license**, **creative commons**, etc.
- pack the texture files and other data before saving that final .blend for publishing
- sign your work, stable email address or website url if possible

The dash prefixes for test-rendering cameras, boxes, floors and lights will help the user know at a glance what to Append and what not to Append from your library.

Here's a quick way to throw out all the stuff you really don't need, including extra meshes, materials and user interface complexity.

- save your current .blend (and make a nice backup file too)
- shut down Blender entirely
- open up Blender again, which will load the default settings
- delete the cube and cameras from the default settings
- Append all the useful parts of your library .blend file (including the *useful* test-rendering items)
- select the proper test-rendering camera for users to try out your model quickly (select, then Ctrl+KEYPAD0)
- adjust the views to ensure important things are visible and ready to render
- save the new library .blend, ready to publish

Blender saves the default settings in a file called **.b.blend** on your disk. If your own preferred default settings are still too complicated for newcomers to understand, you can move that file away temporarily to get the "factory" built-in default settings which the Blender team produces as a part of each new version of the software. Move the file back again when you want to go back to your individual way of working.

220.3. Publishing Your Library

It's helpful to include the .txt file and/or post it separately so that people can read a summary before loading the blend file. This should include any credits, usage notes, layer explanations, and licensing information. For you Unix folks, remember to run it through **unix2dos** to enforce `\r\n` CRLF newlines, readable by people with less flexible tools such as Windows Notepad.

It's also of immense benefit to put up small test-renders of your library objects or materials. They don't need to be large but they should give an honest view of the work you're sharing before a potential user takes the time to download library files that will not be useful to them.

So, where do you publish your work of artistic genius?

- <http://www.e2-productions.com/bmr>¹ Blender 3D Model Repository
- <http://blenderartists.org/>² forums
- <http://www.deviantart.com/>
- <http://www.blendswap.com>
- your own website

If you post things on your own website, try not to rely on a free site that will over-run your bandwidth limits and disappear two months later. Search engine links will sometimes live on for years, just frustrating those who were hoping to find a millimeter-accurate model of the Empire State Building.

220.4. Beyond Libraries

If you have even more time to spare, consider writing up a tutorial on how you achieved any tricky results!

220.5. Thank You

For every artist who chooses to share their creative works with the community, there are a dozen artists, or even hundreds, who thank you immensely.

¹ <http://www.e2-productions.com/bmr>

² <http://blenderartists.org/>

221. Add some depth with stereo

This tutorial contains some tips for how to work with stereo images.

Stereo viewing is to see the same thing from two slightly different angles. This is what humans normally do with their two eyes. There are many ways to view a stereo image. A perfect method is to use a stereo monitor. But they are quite expensive, and many are of very low quality. In the other end of the price-scale you can use crossviewing. Put the two images next to each other, and look at one with one eye, and the other with the other eye. Needs a bit of practice though. A popular solution is red/blue anaglyph glasses, but they give very bad colors.

221.1. The stereo camera

I wanted a stereo camera rig that was easy to work with. It should have three cameras (center, left right), have an easy way to set separation (should be 1/30 of the distance) and I would only need to work with the center cam (position etc.), to others should just follow.

221.1.1. Create it

Manually

1. Reset the cameras position, rotation and size (you can use **[ALT] + [G]**, **[ALT] + [R]** and **[ALT] + [S]**). You may want to note these values first, so you can change them back later. I just make sure there's an IPO-curve for them, for example by making a keyframe (**[I]**).
2. Create two new cameras. Reset their position, rotation and size too.
3. Name them *Camera.Right* and *Camera.Left* or similar. Set LocX to 1 for *Camera.Right* and -1 for *Camera.Left*. (Press **[N]** to see data for selected object, in this window you can change name and values.) (Select a camera by clicking **[RMB]** several times on the cameras, until the right one is selected.)
4. Create a new cube (remember to press **[TAB]** to exit edit mode). Reset position, rotation and size. Name it *Distance Cube* and set these values: LocZ: -30. SizeX: 0.1. SizeY: 0.1. SizeZ: 30.
5. Switch to front view. (**[1]**)
6. Select *Camera.Left*. Then select *Camera* while holding down **[SHIFT]** so both are selected. Press **[CTRL] + [P]** and press **[ENTER]** to register *Camera* as parent. Then press **[CTRL] + [L]** and select *Camera Data*. Repeat with right *Camera.Right* and *Distance Cube*.

With a script

Editor's note

I tried making a script that creates the rig, it almost works. Any help finishing it?

```
import Blender
from Blender import *

# Prepare
scene = Scene.getCurrent()
camera = Object.Get("Camera") # TODO validate that it is a camera
oldLocation = camera.loc
oldRotation = camera.rot
oldSize = camera.size

# Create stuff
c = Camera.New("ortho")
cameraLeft = Object.New("Camera", "Camera.Left")
cameraLeft.link(c)
scene.link(cameraLeft)

c = Camera.New("ortho")
cameraRight = Object.New("Camera", "Camera.Right")
cameraRight.link(c)
scene.link(cameraRight)

dEmpty = Object.New("Empty", "Distance")
scene.link(dEmpty)

# Configure
camera.loc = (0,0,0)
camera.rot = (0,0,0)
camera.size = (1,1,1)

cameraLeft.loc = (-1,0,0)
cameraLeft.rot = (0,0,0)
cameraLeft.size = (1,1,1)

cameraRight.loc = (1,0,0)
cameraRight.rot = (0,0,0)
cameraRight.size = (1,1,1)

dEmpty.loc = (0,0,-60)
dEmpty.rot = (0,0,0)
dEmpty.size = (1,1,1)





scene.update(1)

# Connect
camera.makeParent([cameraLeft, cameraRight, dEmpty], 0, 0)
# do that CTRL+LKEY thing
cameraLeft.link(Camera.Get("Camera"))
cameraRight.link(Camera.Get("Camera"))

# Reset original values
camera.loc = oldLocation
camera.rot = oldRotation
camera.size = oldSize

# Finish
Blender.Redraw()
```

221.1.2. How to use it

- Never change the cube or the two side-cameras. Only change the center camera. Use that one for positioning, rotation etc.
- To set the separation: As always, select the center camera. Resize it (with ) so you can see the end of the cube if needed. Point at the end of the cube with the mouse pointer, and press . Move the mouse pointer to the point of the main motive, that is closes to the camera. The end of the cube may not end exactly there, but that doesn't matter.
- To render (or preview) with one of the side cameras, select it and press  + .

221.1.3. What needs improvement

The cube is visible

Instead of using a cube, I'd rather use something that doesn't render. It could be an "empty", but see below.

The cube is hard to see

It can be hard to see where the cube ends, even in a simple scene. You can select the center camera AND the cube, then it is clearly visible. But you must remember to only select the camera, before you insert a keyframe, so your changes to the cube doesn't get saved.

Distance plane is at infinity

The distance plane is where "zero depth is". When viewing a stereo image, the distance plane is where the medium is. In this rig the distance plane is at infinity, meaning *everything* is in front of it. While stuff popping out in front of the screen is cooler than stuff being "inside" the screen, it's a lot harder to make it look nice. Specially because with everything in front of the screen, it's easy to get stuff that is just way to close to the viewer. It can get so hard to see that the 3d-effect is completely gone.

A different solution is to make the side cameras point at the end of the cube, or add a plane to the rig, and point at that. But then the cameras are no longer parallel, and that creates distortion.

The compromise solution is to (conceptually) render the images too wide, and then crop the excess from the left side of the left image, and the right side of the right image. Then it will look like it points just like in the previously mentioned solution, but without the distortion.

But this give a lot of problems. First, I'd like to do this a smarter way, instead of just following the path describe above. But I don't think that's possible in Blender. (I think povray can actually do this.) So I need to render the image too wide. But if I increase width in output, it actually renders the same width of the motive, but decreases height. Then I need to do some weird math to get the right FOV, and I don't know half the formulas. Cropping must be done outside Blender, and the rest of "the production line" is hard to get back into Blender if wanted.

I guess a couple of planes in the rig close to the cameras could simulate the cropping, but that is of limited value. To make an adjustable distance plane with them seems quite hard to me, it would need some scripting I guess.

221.2. Stereo viewing with the rig

This is where it gets exciting, now you are actually getting something to look at.

1. Create a new screen, you could call it *Stereo View*.
2. The screen should have one big area from side to side.
3. Set the area to *3D View*
4. Unlock it (The *Locks layers and used Camera to Scene-lock*)
5. Select *Camera.Right* and press **CTRL** + **0**
6. Split the area in half so there is two parts next to each other
7. In the *right* area, select *Camera.Left* and press **CTRL** + **0**
8. Adjust zoom in both areas so the frame that shows what is rendered is completely visible, but fill as much as possible. If you can't make them same size, the two areas are not same size.
9. Use cross-view method to see the 3d-effect. Look at the left area with the right eye and right area with the left eye. (It's a skill you need to learn and practice.)
10. Press **SHIFT** + **ALT** + **A** to see your scene animated in stereo.

If you cannot see these, there is a lot of help with viewing them on the web. Use a search engine to find them.

221.3. Stereo rendering

221.3.1. Anaglyph

Create two render layers in the *Render Layers* tab.

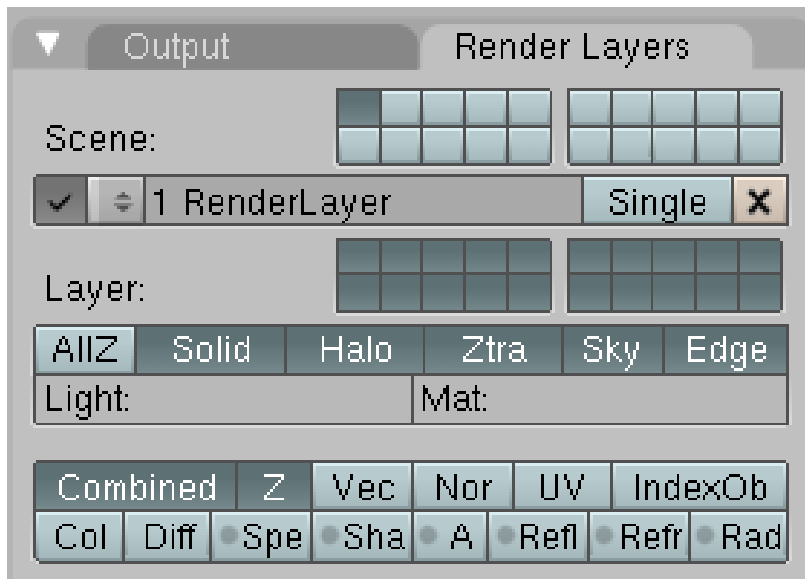


Figure 1430

Editor's note

You need to find how to associate one camera with one render layer

Then go the *Node Editor* and click on the button with a human face. Click on *Uses nodes* and remove the created items by selecting the items and clicking on **DEL**. Now, using the *Add* menu, you need to do this diagram:

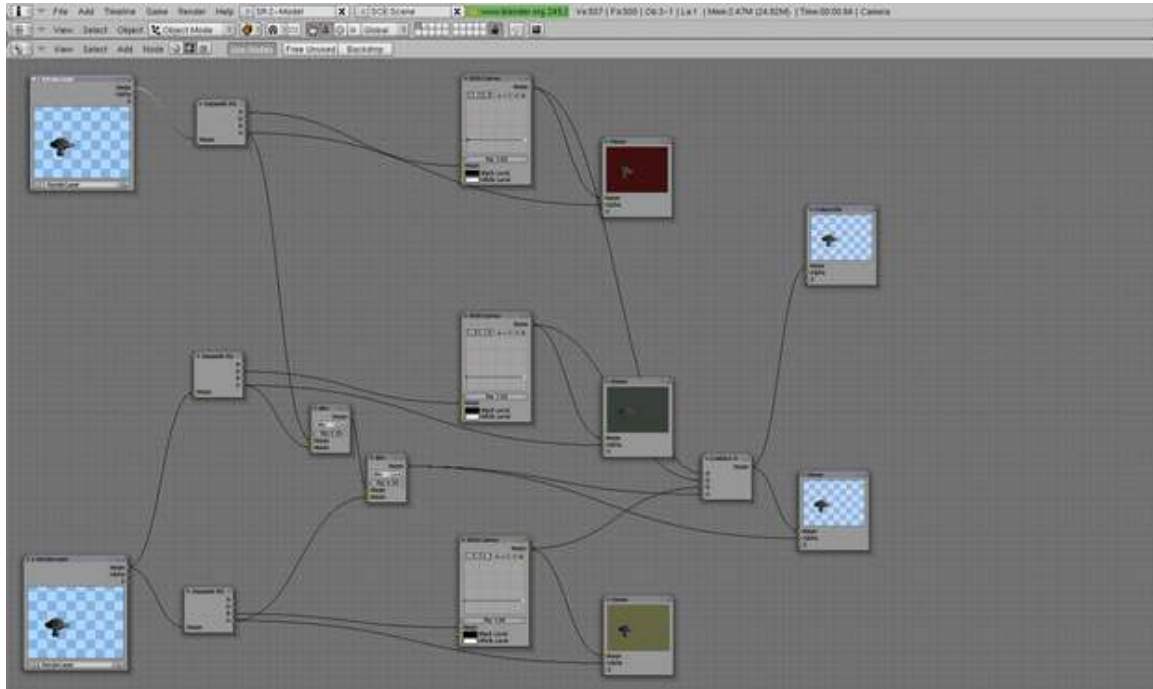


Figure 1431

Once done, return to the *Buttons window* and select the button *Do Composite* on the *Anim* tab:



Figure 1432

Then click on the button *Render* on the *Render* tab:

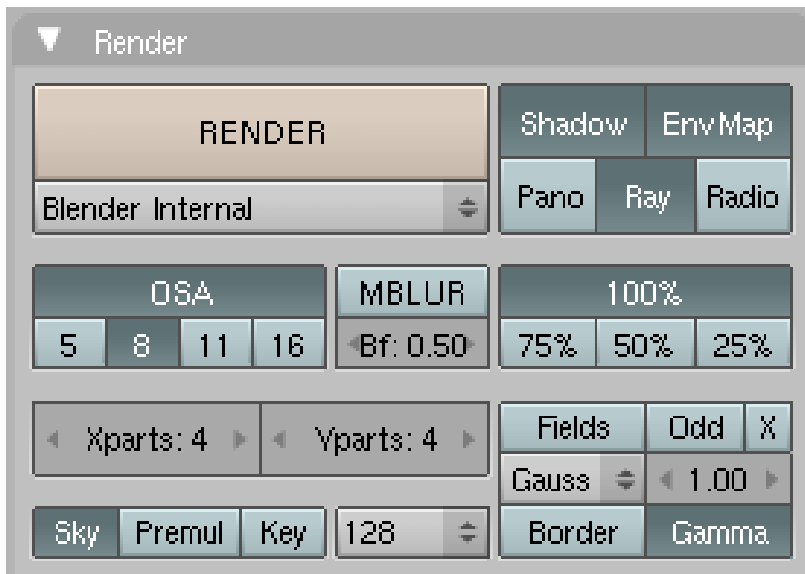


Figure 1433

You should see your objects as an anaglyph:



Figure 1434

222. Fluffy Material

222.1. Spherical Blend Texture

This technique maps a spherical blend texture to the outsides of an object.

Create a new material and name it something intelligent. Create a new texture for this material, change the texture type to Blend, and then in the texture properties, change the blend shape to "Sphere".



Figure 1435

Back in the material buttons, go the "MapInput" tab and change the texture coordinates to Blank-Blank-Z and Nor and as demonstrated in the next screenshot:



Figure 1436

Finally go the "MapTop" tab, and depress the Emit button and set the Texture Blending Mode to "Add":



Figure 1437

That's all there is to it, and here is how it looks:



Figure 1438

222.2. Backlighting

This is a technique that a rather famous blender user called "@ndy" uses. Very simple and effective.

All you need to do is place a Hemi light BEHIND the object you want to light in respect to the active camera.

Breakdown: First add a Hemi lamp to your scene:

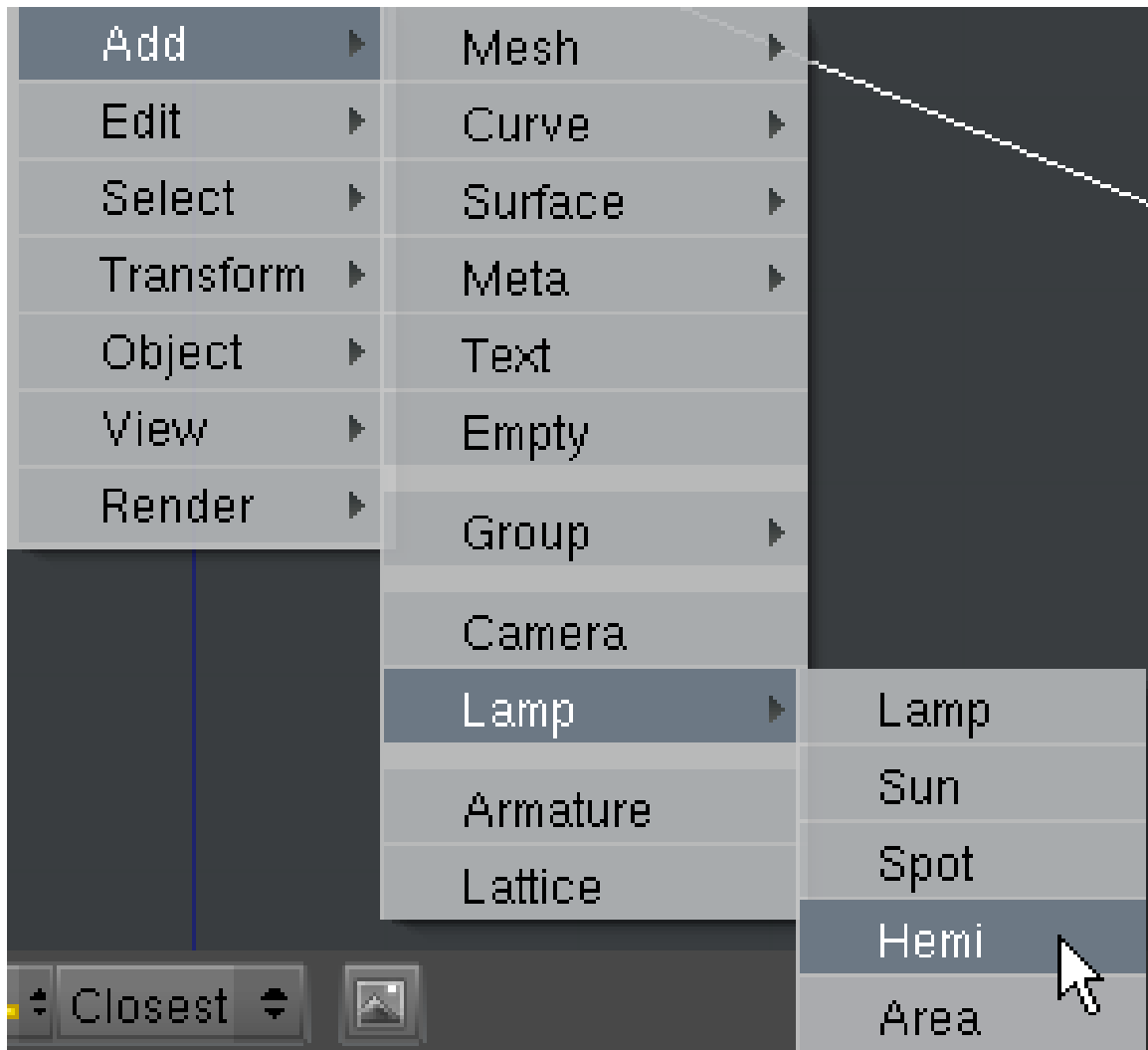


Figure 1439

Next, Select the lamp THEN press and hold down the Shift key, and select the Camera. Press Control-C to bring up the copy attributes menu, and copy location AND rotation.

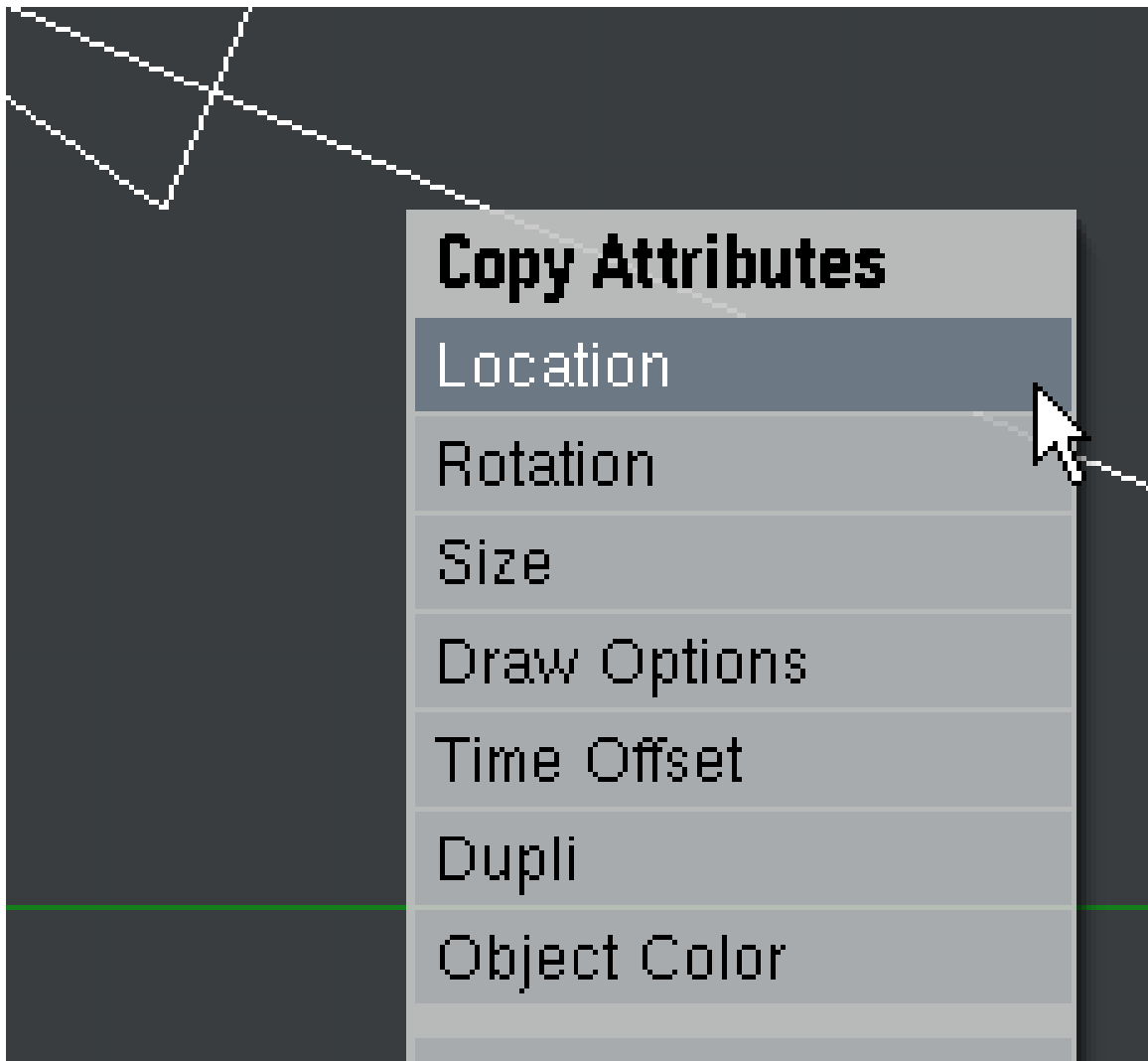


Figure 1440

Now select JUST your lamp, press R to rotate it, then press the X key TWICE to rotate around the local X axis. Using your numpad, key in "180" to rotate the object 180Degrees.

All that remains to do is to press G to grab the lamp, then press Z TWICE to move along the local Z axis, and move the lamp until it is past and behind the object of interest. your resulting setup should look something like this:

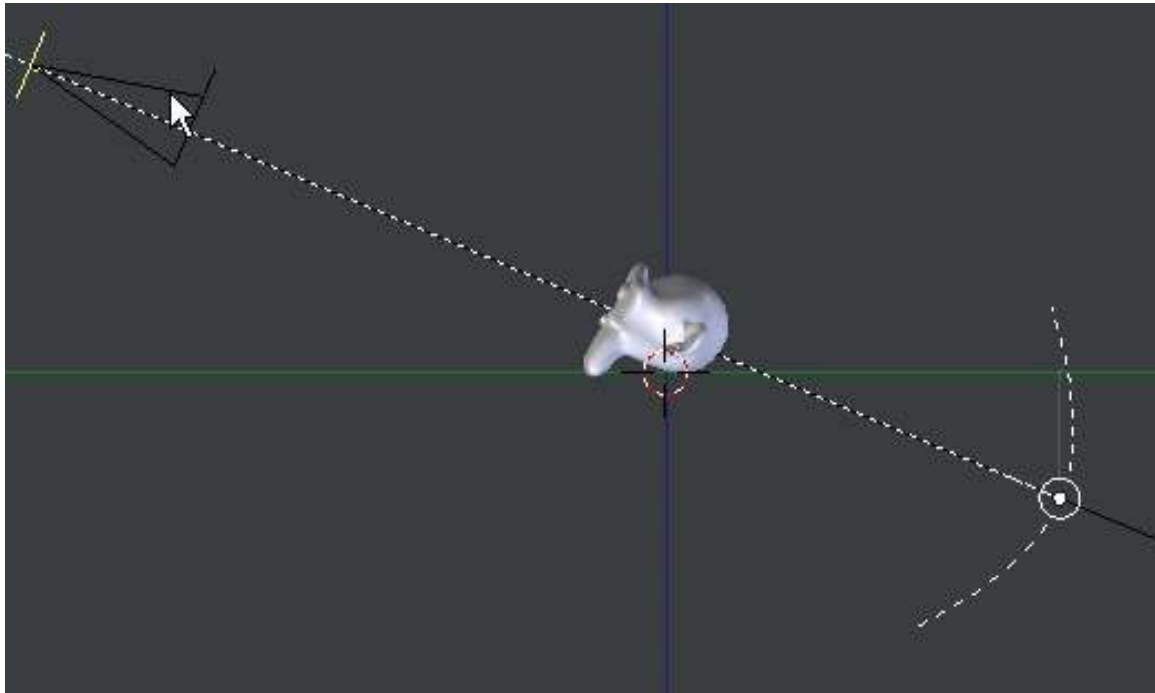


Figure 1441

And here is the rendered result:



Figure 1442

- Color ramp¹ with input set to normal.

Pretty straightforward, but many advise against it.

- Minnaert shader

Available in 2.37, "Darkness" < 1 actually brightens edge. A cool shader, but not very useful for this purpose.

¹ http://www.blender3d.org/cms/Ramp_Shaders.348.0.html

223. Human Body

This tutorial should cover the modelling, skinning and animation of human body, plus facial expressions.

There are some tools to generate human body from parameters like:

- <http://www.makehuman.org/>¹ makehuman² free software program (gpl). The models generated are under MIT license. The models generated can be easily imported in blender.

Use lightwave(obj) export. The exported file is located at ..\Documents\makehuman



Figure 1443

¹ <http://www.makehuman.org/>

² <http://en.wikibooks.org/wiki/%20makehuman>

Delete the original cube first. The human object is smaller than the standard cube. Use the UVImage editor to see the UV mapping but it actually uses a texture stored in the image file texture.png

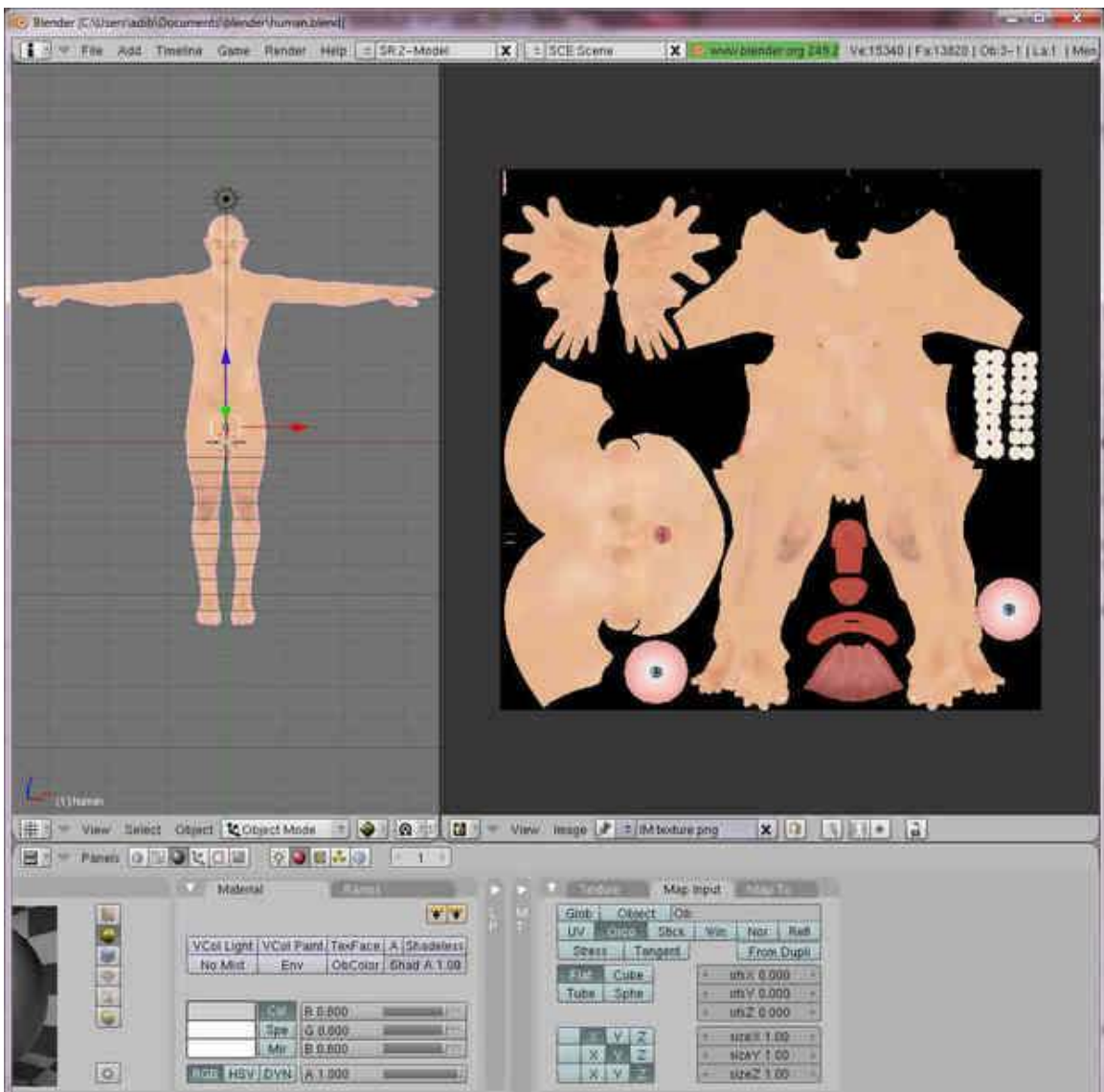


Figure 1444

- <http://www.facegen.com/>³ facegen⁴ proprietary program. Used for wikipedia:The_Elder_Scrolls_IV:_Oblivion⁵.

links to anatomy sites :

- <http://www.fineart.sk/>⁶ fineart⁷

3 <http://www.facegen.com/>

4 <http://en.wikibooks.org/wiki/%20facegen>

5 http://en.wikipedia.org/wiki/The_Elder_Scrolls_IV%3A_Oblivion

6 <http://www.fineart.sk/>

7 <http://en.wikibooks.org/wiki/%20fineart>

tutorials: Blender_3D:_Tutorial_Links_List#Human⁸

⁸ http://en.wikibooks.org/wiki/Blender_3D%3A_Tutorial_Links_List%23Human

224. Using Blender Libraries

You can append libraries in two ways. You can make a local copy of **data blocks** (like objects, meshes, ...) of the content of a .blend file in your .blend file (**appending**) or you can use data blocks from another .blend file (**dynamic linking**).

You can **Append** with the command in the 3d view File -> Append or **SHIFT+F1**. When you give this command a file browser window opens. There are two buttons **Append** and **Link** at the bottom of the window. The default action is *appending*. But you can use *dynamic linking* selecting the link button.

At this point select the .blend file to append. You can select one of the following data blocks type to append:

- **Group**
- **Mesh**
- **Object**
- **Scene**
- **Text**
- **World**
- (not complete yet)

Note that this is a complete list, when you append only block types present in the file will appear. Select the desired type. Now you can select the particular data block to append by selecting its id.

224.1. Indirect linking

When you give an append or link command almost all relations between **data blocks** in Blender get expanded. For example, when you link (or append) a specific Group, all its objects, the meshes associated with the objects, the materials and the animations will be linked (or appended) too. That is called "indirect linking". When you use *dynamic linking* such indirect linked data is not stored when you save a .blend file, when you load the file again blender will look for the indirect linked data blocks in the library file.

224.2. Groups

When you *append* a Group, blender will also create links in the current *Scene* to the objects that are part of the group. The Objects then become visible. However, when you decide to *dynamic link* a Group, it won't do that. To use the objects in your blender project you can use the group as a **duplicator** "**SHIFT+AKEY** -> Group menu -> group id to duplicate".

224.3. See also

- http://www.blender.org/cms/Library_Linking.769.0.html
- http://www.blender.org/cms/Blender_Architecture.336.0.html

225. Beginning Modeling Final Project

Now that you've gotten the hang of 3D modeling, it's important to get some community feedback on your progress. Don't be an idiot and skip this part, or you'll regret it later. Basically this will help you track your progress and give you something that you'll be working on over a long term and something you'll be proud of.

- First, you need to come up with a project idea. You can choose your own modeling project, or choose one from the list below.
- Second, you need to create a model of your idea. Spend a couple of hours on it, and give it some details.
- Third, once you believe you've come far enough with the model, post it in the Works In Progress¹ forum on BlenderArtists.org(formerly elysium.com) (you will have to create an account if you haven't already). Post several screenshots of your model from within the Blender (note: creating screenshots is outside the scope of this wikibook, though see note lower down the page). You can post whatever subject and message with your posting that you would like, or you can use this suggested subject and message:

Subject: Beginning Modeling Final Project - <project name>

Please assist me with any feedback on my model, keeping in mind that I am an absolute beginner still. I appreciate your help.

- Wait for feedback. It usually comes very quickly. If you have any questions about feedback that you are given, don't be afraid to ask your questions in the forum.
- When you and others that have viewed your work feel that you are ready, save your model in some place you can get back to easily. You will continue working on this project once you've learned some new skills.
- Move on to the next page.

(BTW. in Windows and/or maybe other OS, to take a screenshot press 'PrtScn' (PrintScreen). It will copy the screen to clipboard for you to paste in your favourite graphics application. This may not work in other OSs but try anyway. You can also create Blender screenshot directly from Blender using menu **File>Dump 3DView...** or **File>Dump Screen...**)

(In linux under the KDE I use ksnapshot, check under the graphics tab and see if you have it. If not it should be just a google search away :) gl and happy blendering)

(On Mac OS X, press Command (Apple) + Shift + 3 to do a full screen capture)

- List of ideas:

¹ <http://blenderartists.org/forum/forumdisplay.php?s=b6a3696a3b803274bb7c1b49df9a22a2&f=16>

- A Computer and keyboard
- A fishing rod
- A train engine
- A skyscraper
- A robot
- A Tank (real or made up)
- An airplane
- A truck or car
- Household appliances
- A Weapon

226. Using Inkscape to make advanced Bezier curves

226.1. Introduction to Inkscape

Inkscape is a free program that uses SVG (Scalable Vector Graphics) for its file format, and these SVG files can be imported into blender as Bezier curves. Inkscape has some great tools for making advanced shapes that would take forever to make in Blender itself. We will use Inkscape because it is free, and it is very easy to learn.

226.1.1. Installing Inkscape

- *Mac OS X and Windows*: Go to Inkscape.org¹ and click download. there will be installation files for each operating system.
- *Linux*: Go into the terminal.

For Arch Linux, type **pacman -S inkscape**

For Debian based distributions, type **sudo apt-get install inkscape**

For Fedora and rpm based distributions, type **yum -y install inkscape**

For any other distribution, check your package manager to see if it has a package for inkscape. if not, download the source code at Inkscape.org² and run PKGBUILD in the directory of the source.

Inkscape is a pretty large program, so it will take a while to download and install.

226.2. Getting Started

Because blender doesn't import any effects or filters from SVG files, we will only have to learn basic path modeling. This is very easy to learn, and will take no time at all. Once it's finished installing, open Inkscape. You will see a large empty page and a bunch of buttons to the left. Don't mind the empty page, we won't be needing it. We will re-size this once we are done with the tutorial. Before we get started, I will show you how to use some of the controls. You will probably be right clicking a

1 <http://inkscape.org/>

2 <http://inkscape.org/>

lot because you're used to blender, but that's okay. Let's start out with a star to practice some of the controls. Click the Star/Polygon tool to the left, and click anywhere on the paper and drag.

226.2.1. Simple Controls

go back to "Object Mode" by clicking the cursor icon to the left. Now, let's try out some of the controls:

Action	Hotkey
Zoom In/Out	CTRL+SCROLL
Move along Y Axis	SCROLL
Move along X Axis	SHIFT+SCROLL
Move along both axes	Hold MMB and Drag
Select Path/Curve	LMB
Select All	CTRL+A
Select all paths/curves within a rectangle	Hold LMB and drag (Like selecting icons on your desktop)

These are only a few of the controls, you will learn more controls later in this tutorial.

226.2.2. Editing your Bezier path

Go back to the Star/Polygon tool so we can edit the star. You will see 2 dots appear on the star. You can drag these around, and see what happens. Another thing you can do is add more points to the star, change the spike ratio, make it rounded, and make it randomized with the "CHANGE:" Dialog right above your image.



Figure 1445

If you want to edit each node separately without symmetry, go back to 'Select and Transform Objects' mode

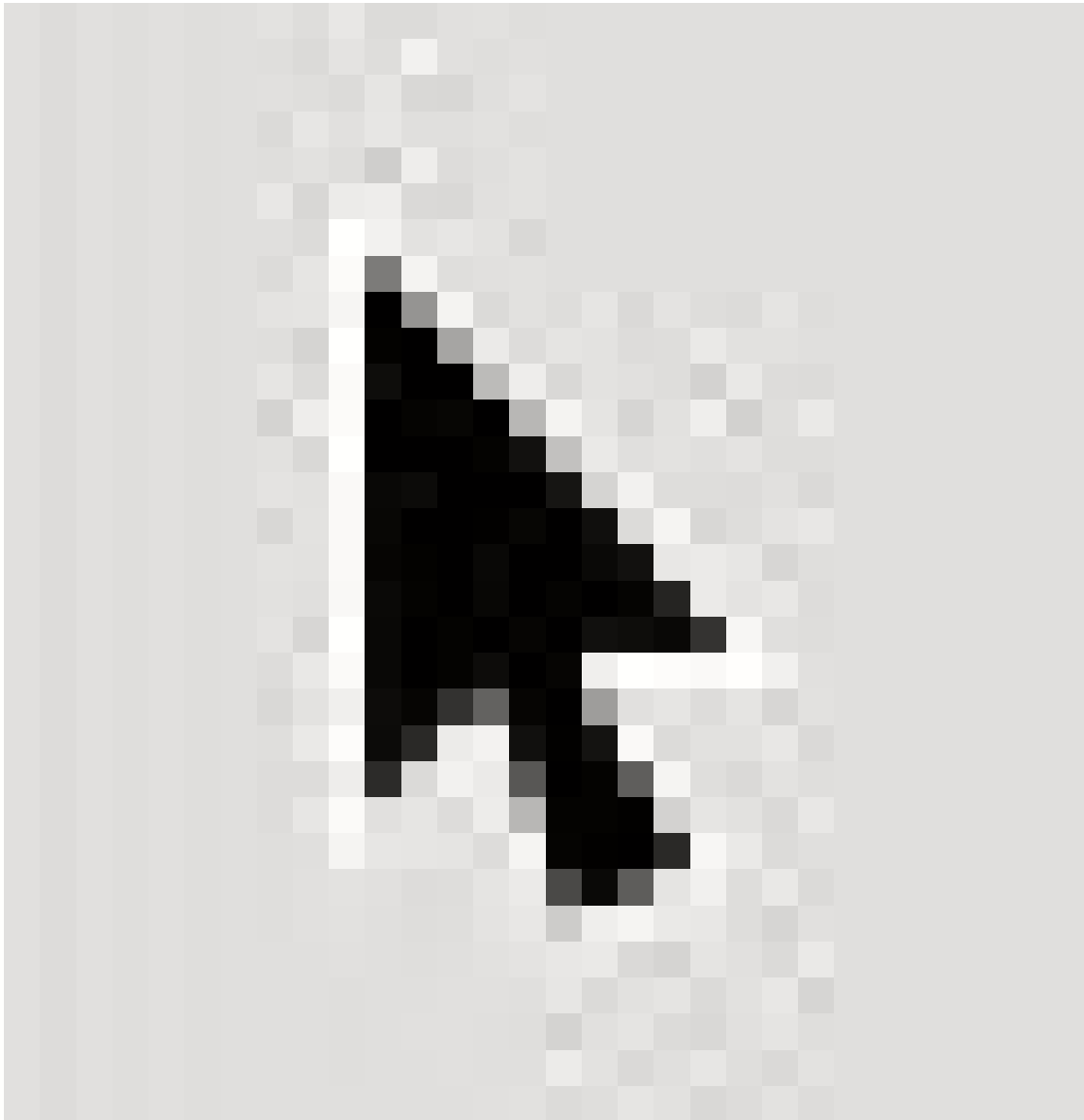


Figure 1446

and click Path>Object to Path. Now you can just go to 'Edit Paths by Nodes' mode

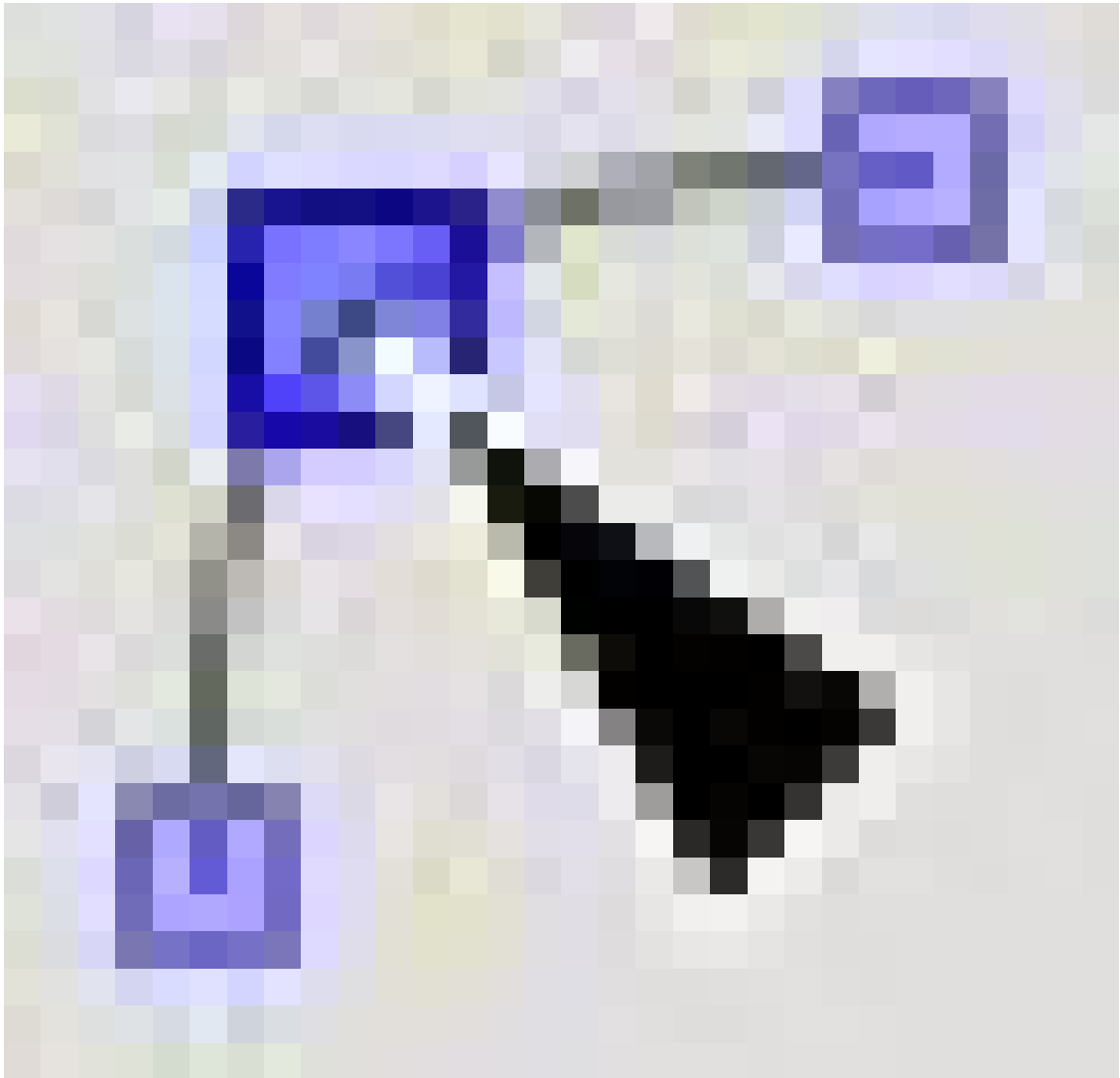


Figure 1447

and play around with the shape a bit.

By clicking and dragging, you can move a node or curve an edge. when you make a curve, 2 nodes will show up that you can drag and edit the curve with more precision.

To add a node in the path, select an edge and press the plus button

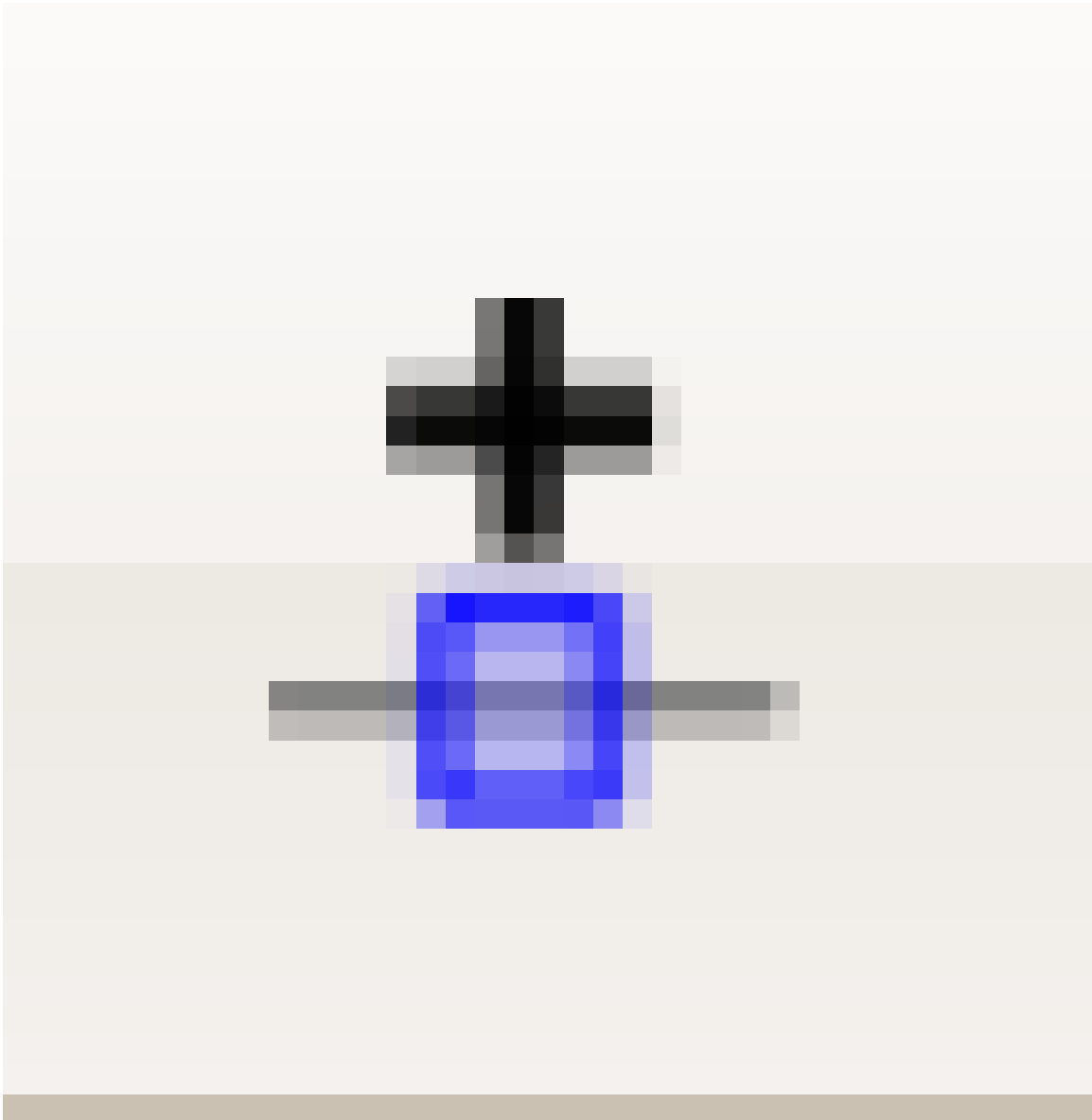


Figure 1448

to remove a node, press the minus button, and, well, the buttons pretty much explain themselves, so you can just play around and see what you get.

After you think you know how to edit paths pretty well, let's try to make a path from scratch. Click the 'Draw Bezier Curves' button

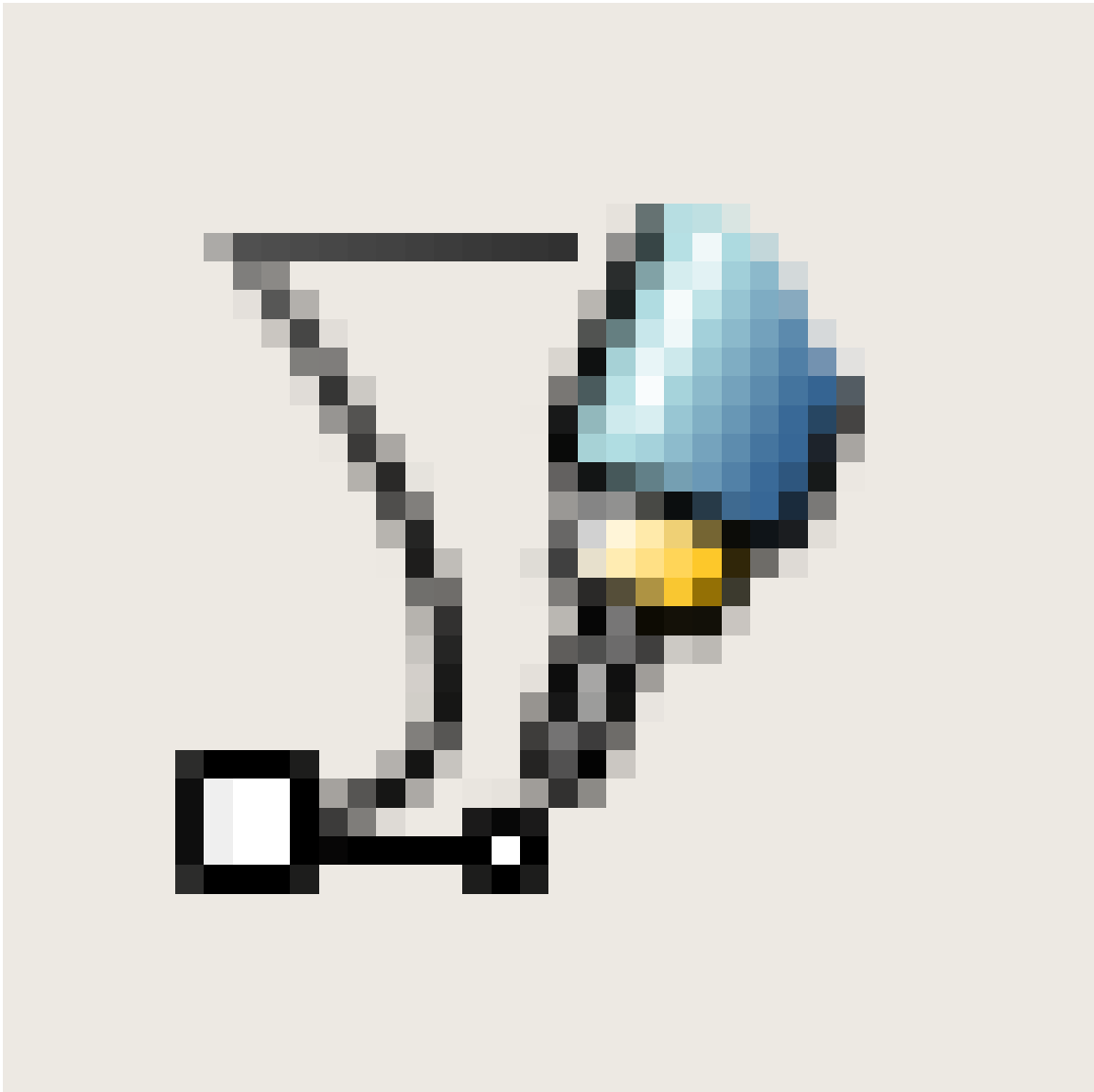


Figure 1449

and click to add nodes for the shape you want. If you click and drag, you will make a node and curve the node while dragging the mouse. To finish the path, go around to the first node that you made and click on it. This will finish the path and make it a solid shape. When it is done, a outline, or stroke of the shape will appear. If you want it to show as a solid shape, you can click one of the colors on the bottom and it will fill the path with that color.

Now, move the star inside the path (In 'Select and Transform Objects' mode, of course), without it overlapping the borders of the path. If the star is too big, scale it down and hold ctrl to keep the same scale on the X and Y axis. Select both your path and the star, and press Path>Difference to cut a star-shaped hole inside your path.

Congratulations, You have made a completely pointless random shape and learned the basics of creating a shape in inkscape! Now, It's time to get this into Blender! Save your SVG image, and open up blender. Press File>Import>Paths(.svg, .ps, .eps, .ai, gimp) (Blender 2.49). Select your SVG

file and the settings are fine how they are. Press OK and see your path appear in blender (You may have to delete the default cube to see it) From here, you can Extrude, bevel, and do whatever you want to you bezier path in blender.

The result may look something like this:

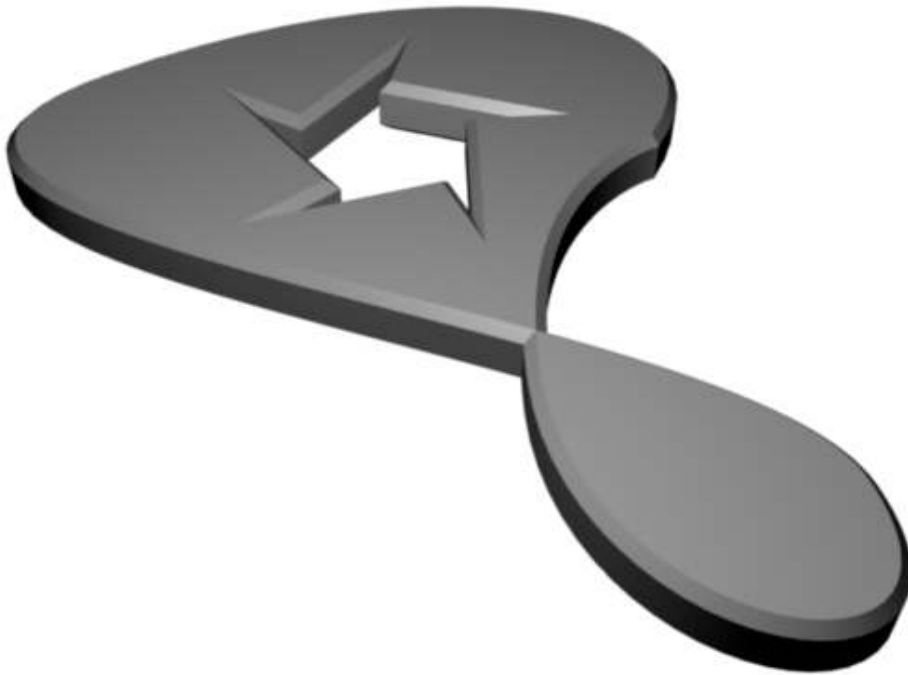


Figure 1450

227. Platonic Solids

The **Platonic solids** or **Platonic polyhedra** are the convex polyhedra where all faces are copies of the same regular polygon, and the same number of edges meet at every vertex. There are five of these shapes: the tetrahedron (like a pyramid but with a triangular base): cube, octahedron, dodecahedron and icosahedron. Some of these figures ships along with Blender by default. Others can also be built using simple grabs and scales.

227.0.3. The Tetrahedron

Bring up the Add Mesh menu, and select a Cone. Set the number of Vertices to 3, leave the Radius at its default 1.0, and set the Depth to $1.41 \doteq \sqrt{\frac{8}{4}}$.

227.0.4. The Cube

This happens to be a built-in shape in Blender. Just bring up the Add Mesh menu, and select Cube. Done!

227.0.5. The Octahedron

This shape is the *dual* of the cube—it has vertexes where the cube has faces, and faces where the cube has vertexes. To make it, first create a cube. Press **[TAB]** to switch to Edit mode. All the vertices should already be selected. Press **[W]** to bring up the Specials menu, and select the Bevel function. As you move the mouse, you will see each vertex of the cube turn into a triangular face; simply keep moving the mouse until these faces grow to their maximum extent and meet up with each other; at that point, press **[LMB]** to finish the operation. Finally, bring up the Specials menu again, and this time Remove Doubles.

227.0.6. The Icosahedron

Bring up the Add Mesh menu, and select an Icosphere. Set the Subdivision to 1. Simple!

227.0.7. The Dodecahedron

This shape is the dual of the icosahedron. To create it, make an icosahedron as above. Then do what you did to make an octahedron out of a cube: press **[TAB]** to switch to Edit mode. All the vertices should already be selected. Press **[W]** to bring up the Specials menu, and select the Bevel function. As you move the mouse, you will see each vertex of the icosahedron turn into a pentagonal face;

simply keep moving the mouse until these faces grow to their maximum extent and meet up with each other; at that point, press **LMB** to finish the operation. Finally, bring up the Specials menu again, and this time Remove Doubles.



Figure 1451

227.0.8. Exercise

What's the dual of the tetrahedron? Try applying the Bevel operation to one of those; what do you end up with?

227.1. External Link

Wikipedia article¹

¹ http://en.wikipedia.org/wiki/Platonic_solid

228. Polygonal Modeling

As the name suggest, polygonal modeling involves constructing the models out of polygons. Strictly, software packages works only with triangles. Blender could show you quads and some software packages could even let you work with polygons with more than 4 sides (n-gons) but these packages work internally only with triangles. These triangles are hidden so that you could concentrate more on modeling.

While different modelers works differently, they would still follow certain workflows (some call this techniques). These are familiarly called box modeling and polygon-by-polygon modeling (poly-by-poly modeling).

Box modeling is a top-down approach where modelers start with a primitive (usually a cube (box), hence the box in box modeling). Then from this primitive, the model's form is build up where details are gradually added.

Poly-by-poly modeling is almost the opposite of box modeling. It is the bottom-up approach where modelers start with a plane or even a vertex. Using extrusion and other tools, more geometries are added. Polygon by polygon the model is gradually build.

None of the two is better than the other but there are instances where one workflow lends much better to the situation. One typical advantage of box modeling is that at early stage, you are able to conceive first the form (the whole) of the model and is better equipped to do general corrections without bothering yet with the details. Poly-by-poly otherwise give you more control on the geometry and is much more easier to use in modeling complex forms like the human ear. Complications do arise when working with the two. In box modeling, it is typically difficult to manage the geometry when adding details. Experience is necessary so as not to create a mess. Poly-by-poly modeling otherwise has its own quirks. Since you are going detail by detail, it is easy to make a mess with the models form (i.e. wrong proportions).

While the two are opposites, they compliment each other beautifully. Most modelers would combine the two workflows in a variety of proportions, combining the strength of each. For example one might model a human body using box modeling while the head and the ears are done using poly-by-poly modeling.

Blocking with primitives is added to enrich approaches is modeling. It is basically box modeling in conjunction with divide-and-conquer approach. Different parts are modeled separately with their own primitives and later attached with the others to form the whole.

229. Box Modeling

Easy to undertake and flexible, box modeling is a favorite of beginners and veterans alike. It is fun to work with and general results are visible in a short time. It is a powerful work flow that any modelers should know.

229.1. Before We Start

Before doing any modeling, it is vital to plan first. Gather references and make a general plan on how to tackle the modeling phase. Being ready will save you a lot of complications later

229.2. The Work Flow

With box modeling, start with a primitive that is appropriate for your subject. Most start with a box and generally a good primitive to start. Then, from the primitive, using a variety of tools, mold the essential form of your model. Also called the roughing phase. Don't delve on details here, those are to be tackled later. Finally, on this form, we go to the nearly recursive process of adding details. Layer be layer, details are added until the required amount of detail is achieved.

Start with a Primitive

p The Primitive

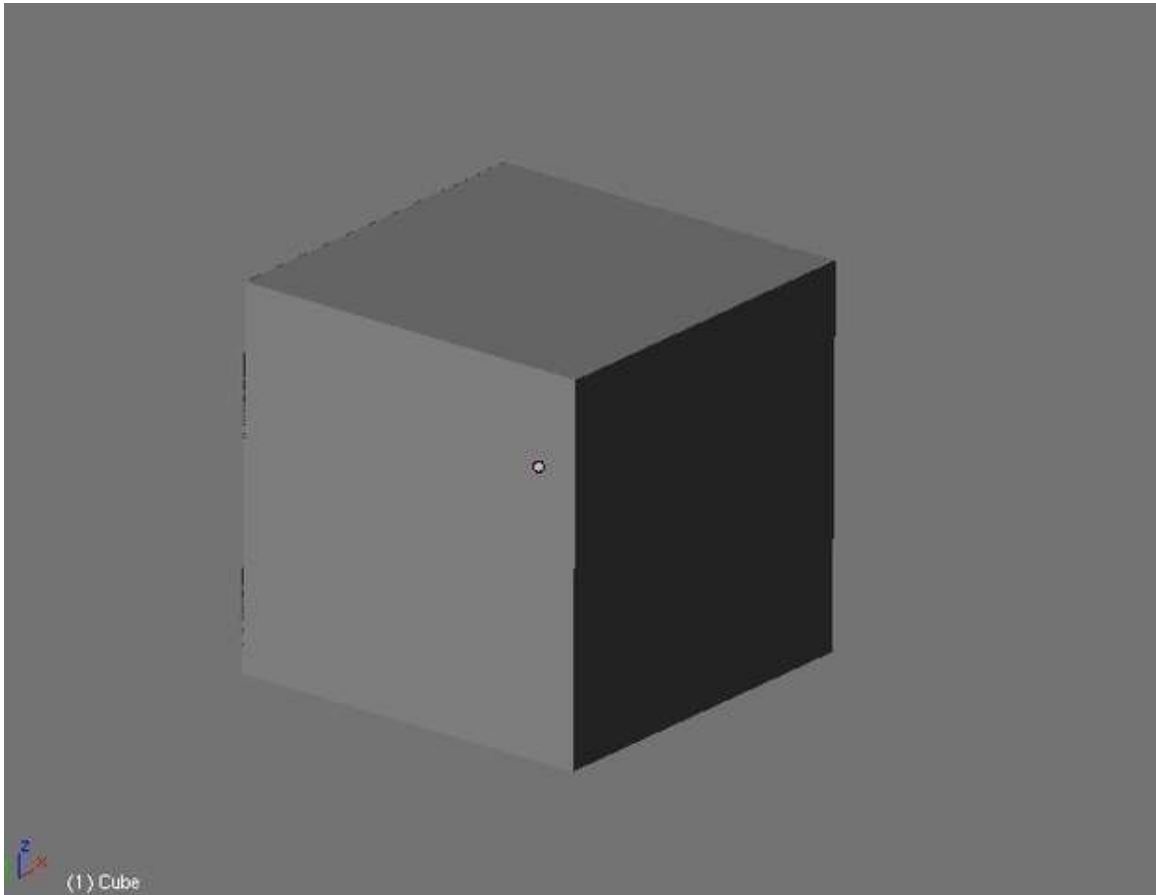


Figure 1452

Select the proper primitive for your work. Most modelers start with a cube. The cube is the most flexible primitive available and is suitable to almost all form of subjects. But, in many cases, selecting other form of primitives would cut out most of the modeling required. A gun is a cinch to model using a cylinder. With a little adjustment, a torus forms a good doughnut. In an instant a ball can be made out of a sphere. So produce your primitive to work with.

Rough It Out

p The Form

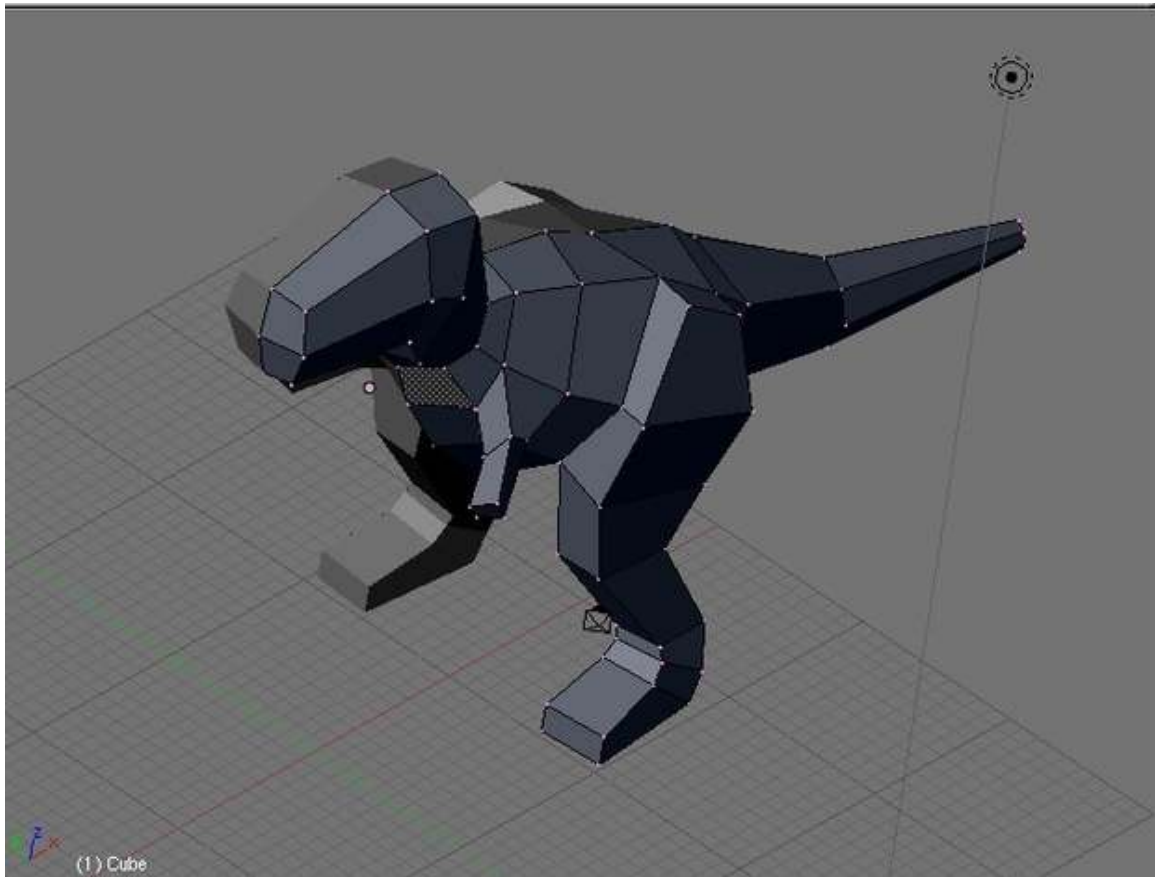


Figure 1453

Take the primitive and start modifying it to capture the essential form of your subject. Tools like extrude, loop cut, scale and grab are very handy for this. Avoid worrying about detail at this stage. Those are to be done.

Adding Details, Details and more Details

p Details, details and more details

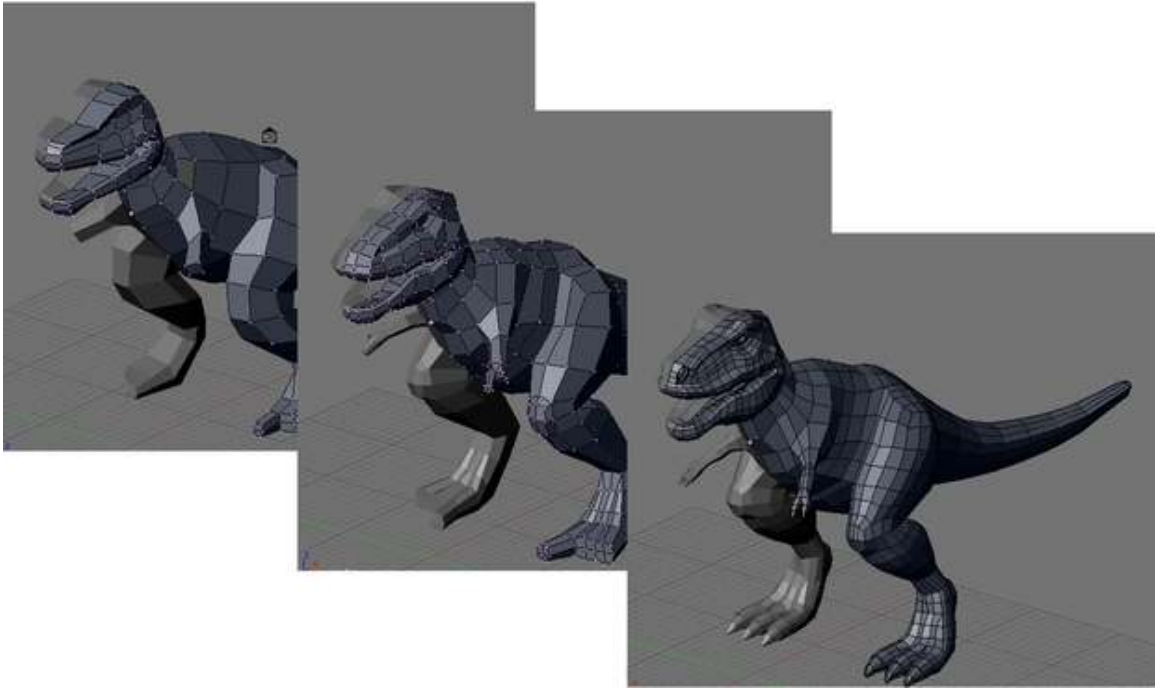


Figure 1454

Now, we approach the most difficult and unquestionably the most fun part. Using a variety of modeling tools, modify the form again to incorporate details. Add details layer by layer, from general to specific. Planning, studying and experience would help you go through. Continue until you achieve the desired level of detail.

Adding details requires subdivision or addition of polygons. The knife and the loop cut tool is handy for subdividing meshes. Extrusion is great for adding details like horns and fingers. Face and edge loop must be herded to other directions at times and warrants a study by itself. Blender provides you with required tools to achieve all of this.

At this stage, beginners and experienced modelers alike will find their meshes getting more and more difficult to work with as the mesh gets a more and more dense at each level of details added. Planning, an eye to the edge flows and experience alleviate this. So plan and keep practicing.

230. Illustrative example: Model a Chair (Swan Chair)

Now let's start with real modeling. In this tutorial we will study in greater depth how to model using box modeling work flow. Let start with a swan chair. It is great piece to model for its a popular piece of furniture so reference images are easy to find. It also have a simple geometric structure so its shape easy to grasp and also contains simple edge and face loop flow so as not to necessitate difficult edge loop face loop flow herding techniques.

p Scene with the chair model as seen in Blender

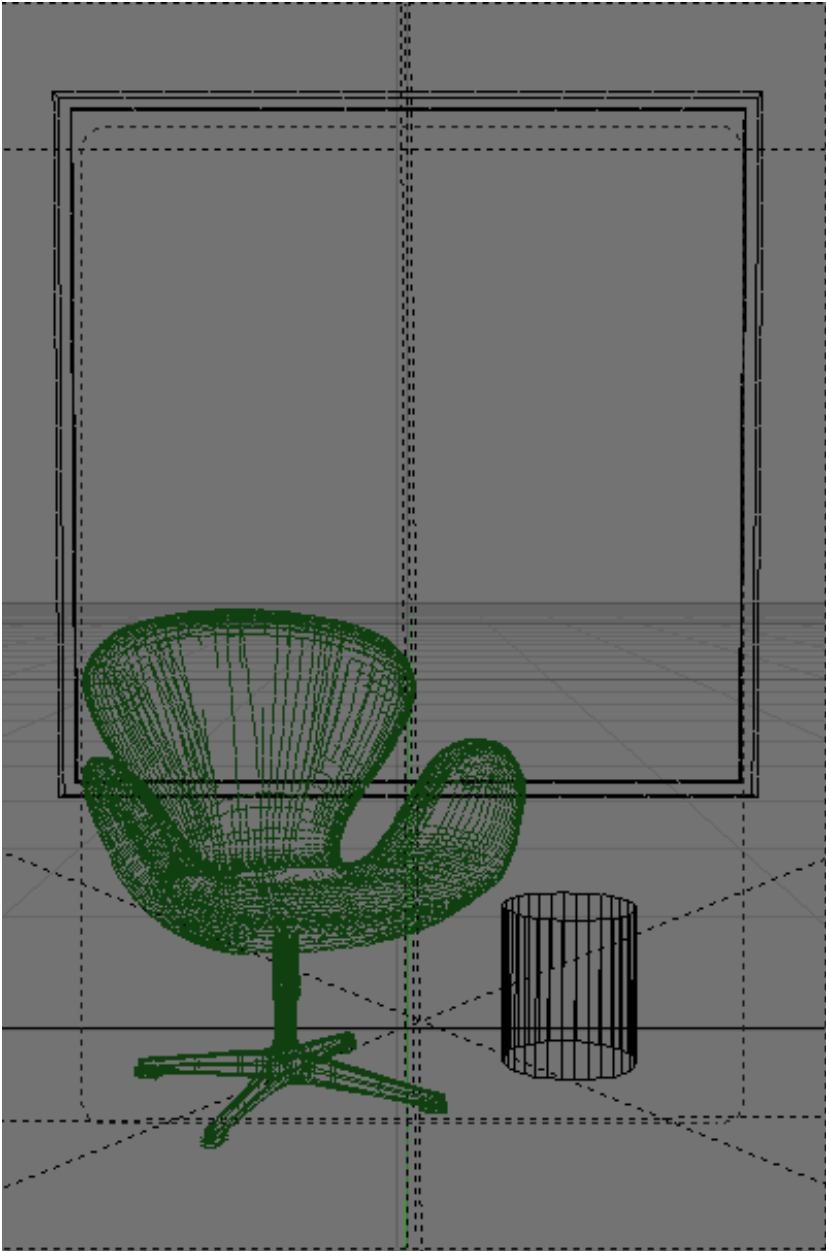


Figure 1455

p Rendered Scene with the chair



Figure 1456

The techniques learned could also be used to model other chairs (Example: the Egg chair, Tulip chair or Parson's chair which is advised to be modeled too) and other subjects.

231. Model a Chair-Preparations

Many people do not know what a Swan chair look like. Its time to open up the internet browser and search for images. Or if you have an actual Swan chair with you, your lucky. You can look at a real model while modeling. This would simplify decisions and remove the need for guessing certain details that you are not available with image searches alone.

231.1. Knowing our model

We need to know what we are modeling look like. You can't do this without references to guide you. A simple image search with Google would provide you the references you need.

Gather several images that gives you different views of the chair. A front and a side view is great to use as a background image to guide as while modeling. Also, gather at least one image that shows the chair at an oblique angle so that it will provide a much more "3d" view. This will shows more clearly the shape of the chair, more than the front and side view alone would give you.

231.2. Loop study

Take time to study the model. For simple models like this, it would be quite quick, especially with experience. For this project, the loops are as shown bellow.

p Swan chair's loops

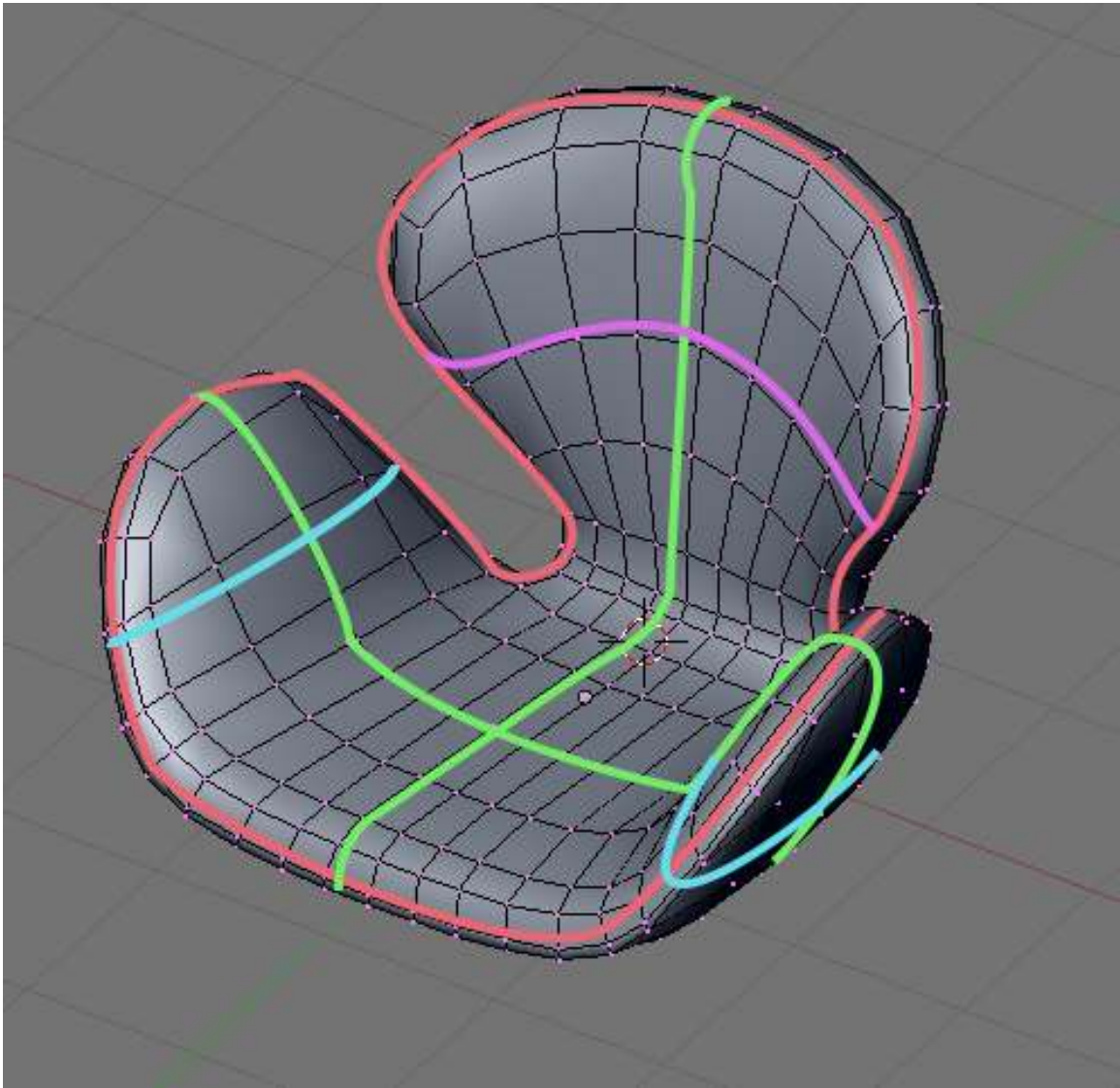


Figure 1457

The pink loop goes around the edge of the chair defining the chair's edges and thickness. The light green, light blue and light purple loops follows and create the form of the chair. The crossing green loops forms the bottom of the chair. The light blue and green loops contribute for the form of the "wings", while the purple and green loops form the back. Its a simple loop structure, its topologically (sorry for using the term) the same as this cube extruded to form a T (upside down so the loops would match more).

p The T shape piece loops

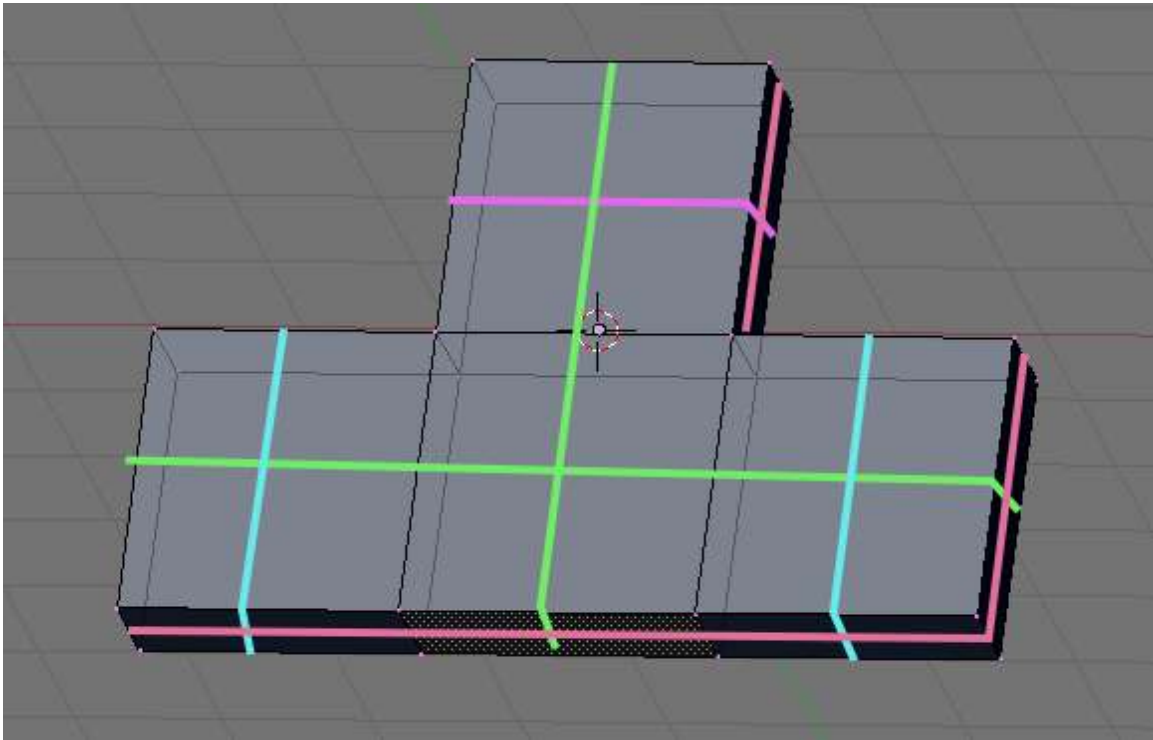


Figure 1458

While modeling keep these loops in mind and try to achieve the loops. For this project, the loops are achieved with simple extrusions of a cube, just like the T piece above, but in a form that looks more like the chair.

232. Model a Chair-The Seat

After studying the subject, we are ready to tackle modeling it. Start up Blender.

232.1. Start with a Primitive

For this project we would use a cube. The cube already has several face loop we need and none of that we don't, so this primitive is a good starting point.

Opening Blender, we are already supplied with the default cube which is handy. If you don't have the default cube because you change the setting or remove it, simply add a new cube. Name this primitive something sensible like "Swan Chair" because it would become our chair later. Move the cube or adjust the background image until the cube is centered at our image. After doing that we proceed shaping our cube to create the basic form of our chair.

232.2. Rough it out

Since we are modeling a symmetrical object, lets take advantage of it and use the mirror modifier. Cut the cube in the front exactly in half by using loop cut at center (Shift + RKey -> middle click, or Ctrl + RKey -> middle click on this editor's computer). For those with no middle buttons you could use subdivide. Select the vertical edges (go to edge select mode first, Ctrl + Tab -> 2) then use subdivide tool (WKey -> 1). Remove the left half by deleting the vertices. Then add a mirror modifier in the modifier stack. This is available at the edit buttons, select the modifiers tab the click the "Add Modifier" button and select "Mirror" from the selection. Activate "do clipping" from the mirror modifier panel that appeared to prevent accidental movement of the central vertices.

(Noob Note: If you follow these instructions, you'll have a gap between the mesh and its mirror. I fixed this by increasing the "Merge Limit" option in the Mirror modifier menu to .1.)

p The cube primitive with mirror modifier on

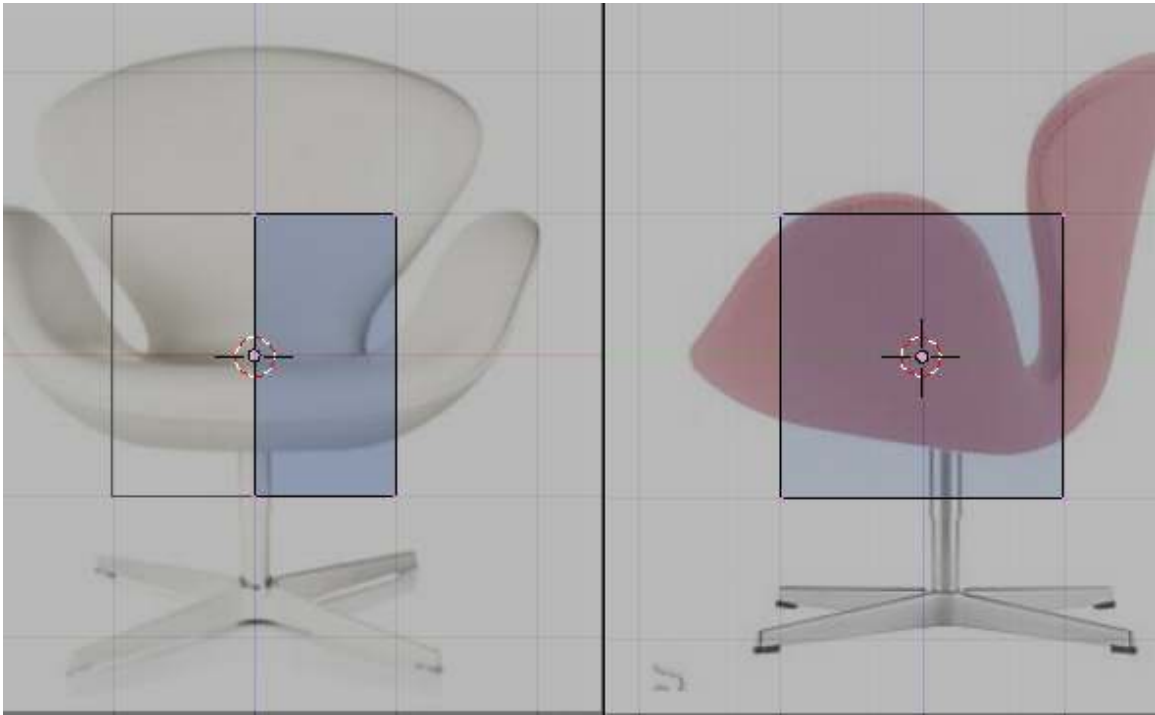


Figure 1459

Move the vertices so that it would follow the shape of the bottom of the chair image.

p Shaping the bottom

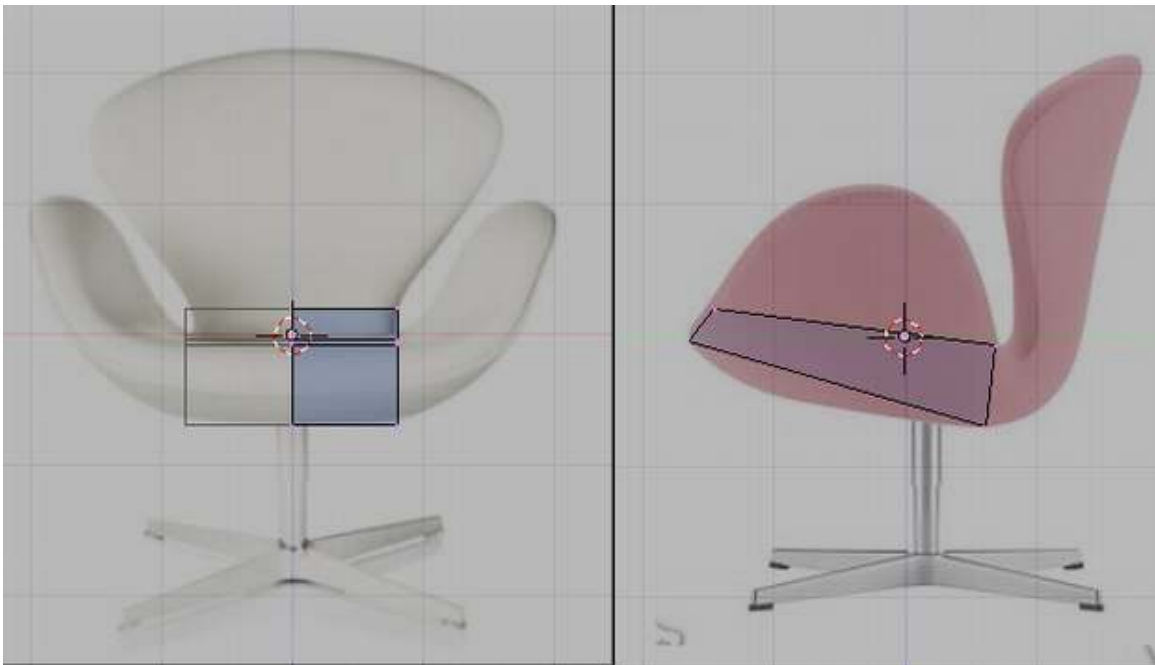


Figure 1460

Note that the reference images used does not match very well. This is because of perspective and camera angle distortion. In this case use the side image since it shows less distortion. Use the front reference image as an eyeballing guide for the front side of the chair. An oblique reference as stated earlier in planning stages is helpful to guide you in this condition.

Add an edgeloop using the loop cut tool (Shift+RKey) as shown in the image bellow. Move the newly created vertices to follow the references

p Shaping the bottom

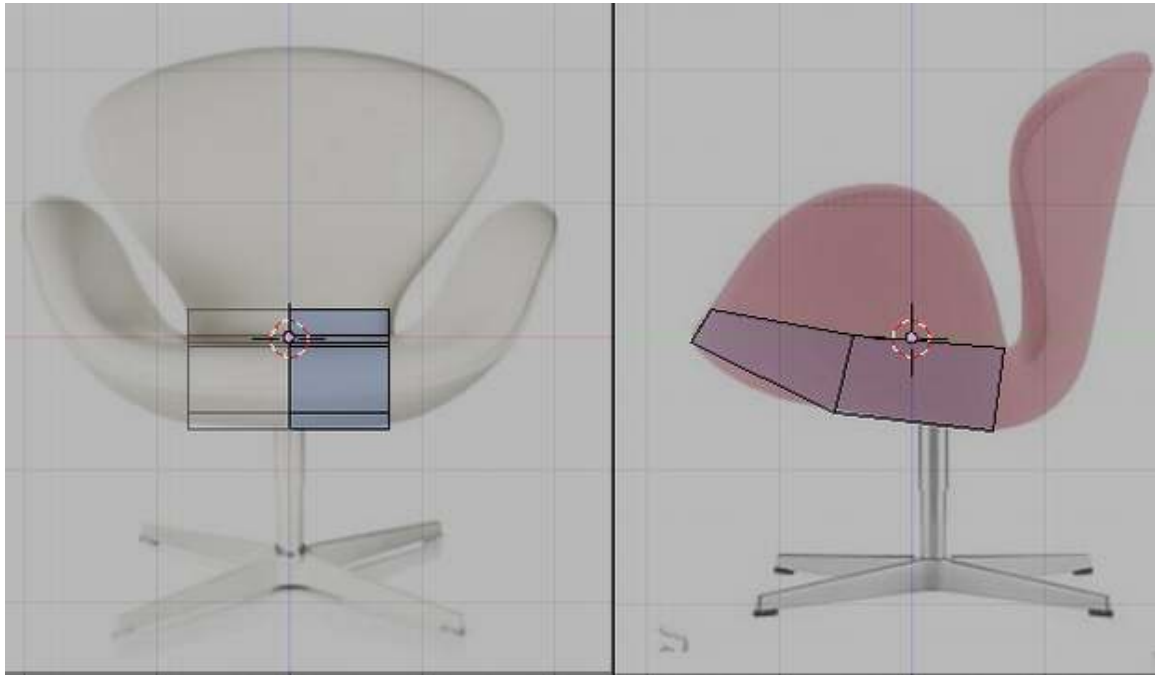


Figure 1461

Its is good practice that while modeling you had to view your model at various views not only the standard front, side and top views. Remember that you are modeling a 3d object. What looks good in this some views are not necessarily good in other views. This would result in complications like the familiar flat face look in face models. Moving your view around would help you prevent this. Shown bellow is the view of the model at an oblique view.

p Shaping the bottom oblique view

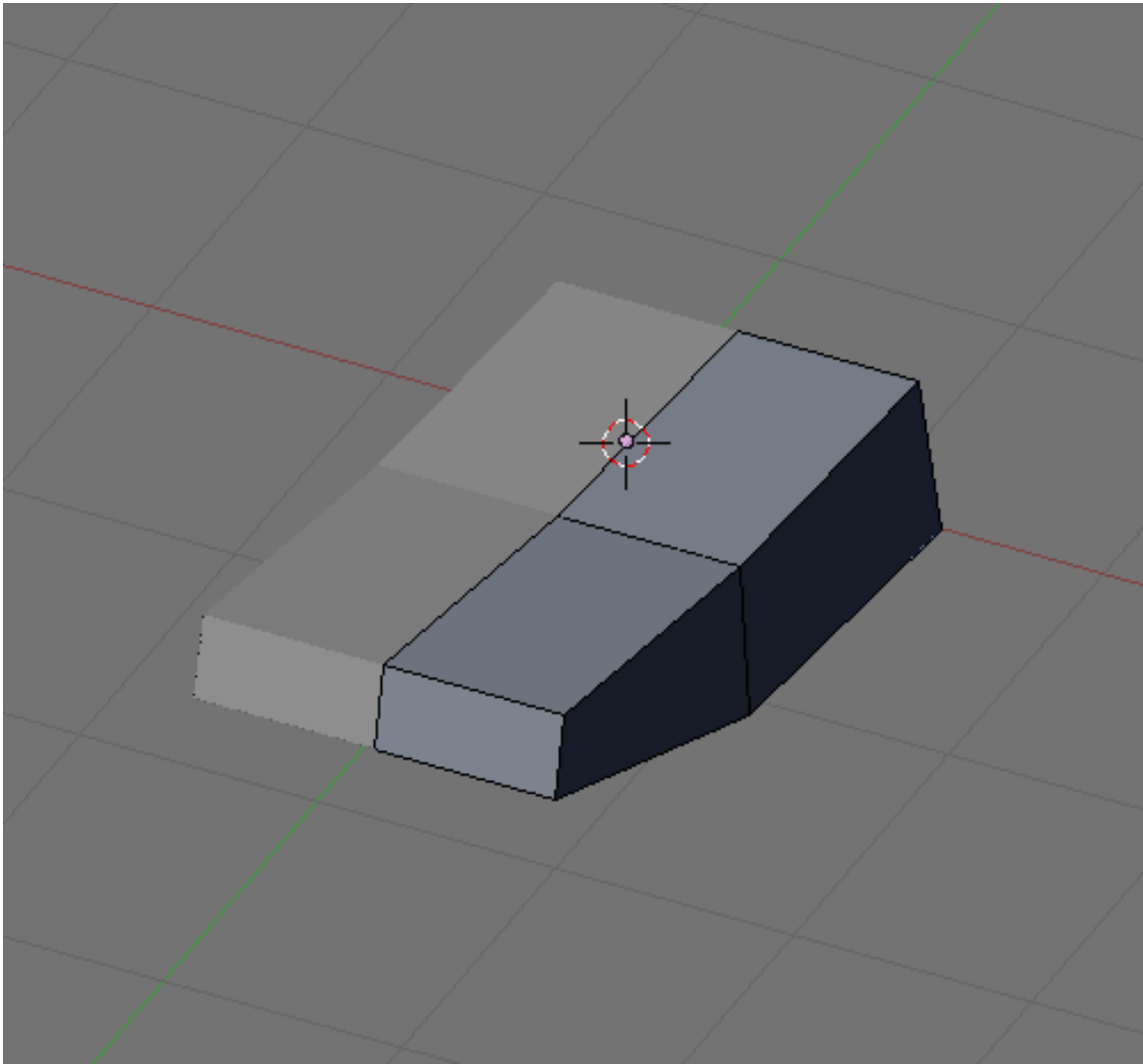


Figure 1462

Now extrude the back part of the model to form the back part of the chair. Adjust the newly formed vertices to follow your guide images as shown bellow.

p Shaping the back

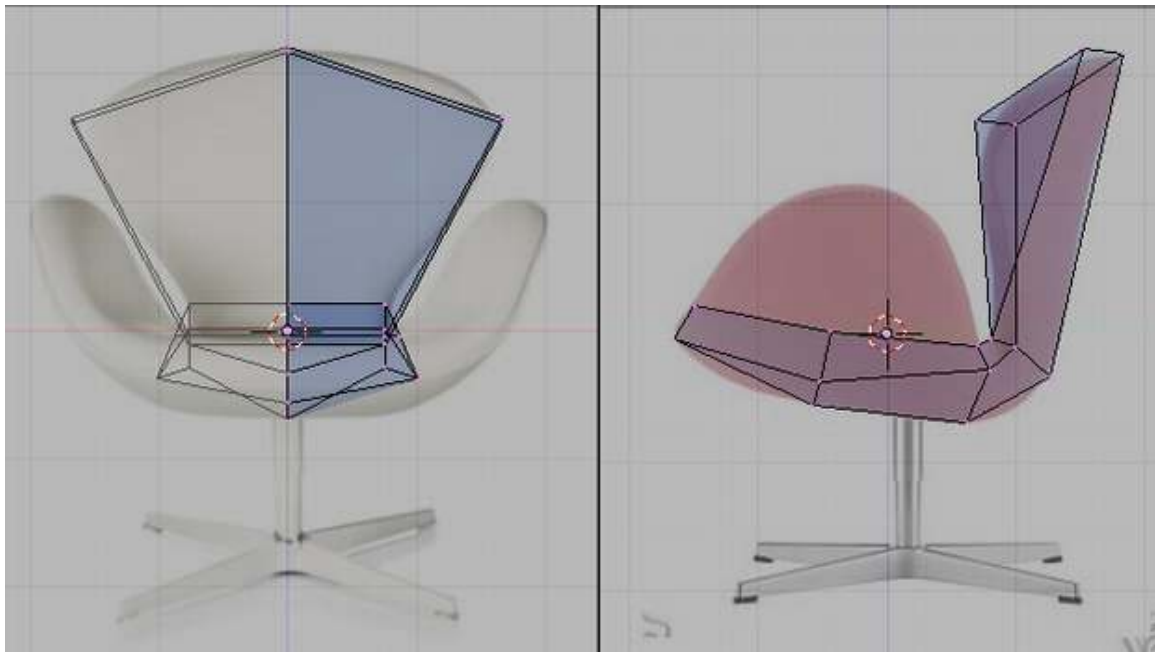


Figure 1463

Its a good practice to see the model at various views while modeling here is how it look like now at an oblique view:

p Shaping the back oblique view

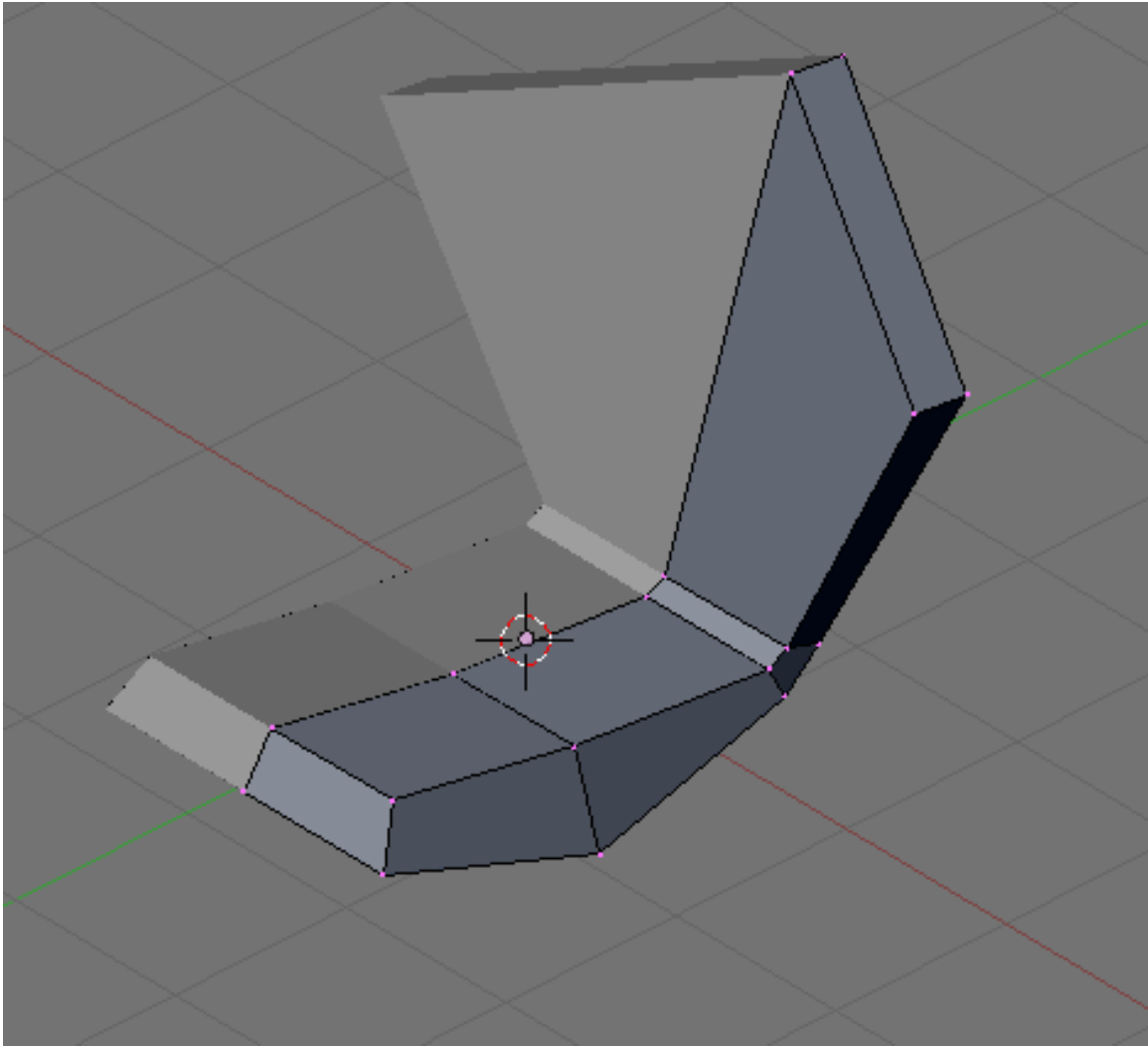


Figure 1464

Select the following faces. We are going to extrude it to form the sides for the chair.

p Shaping the sides, faces to select

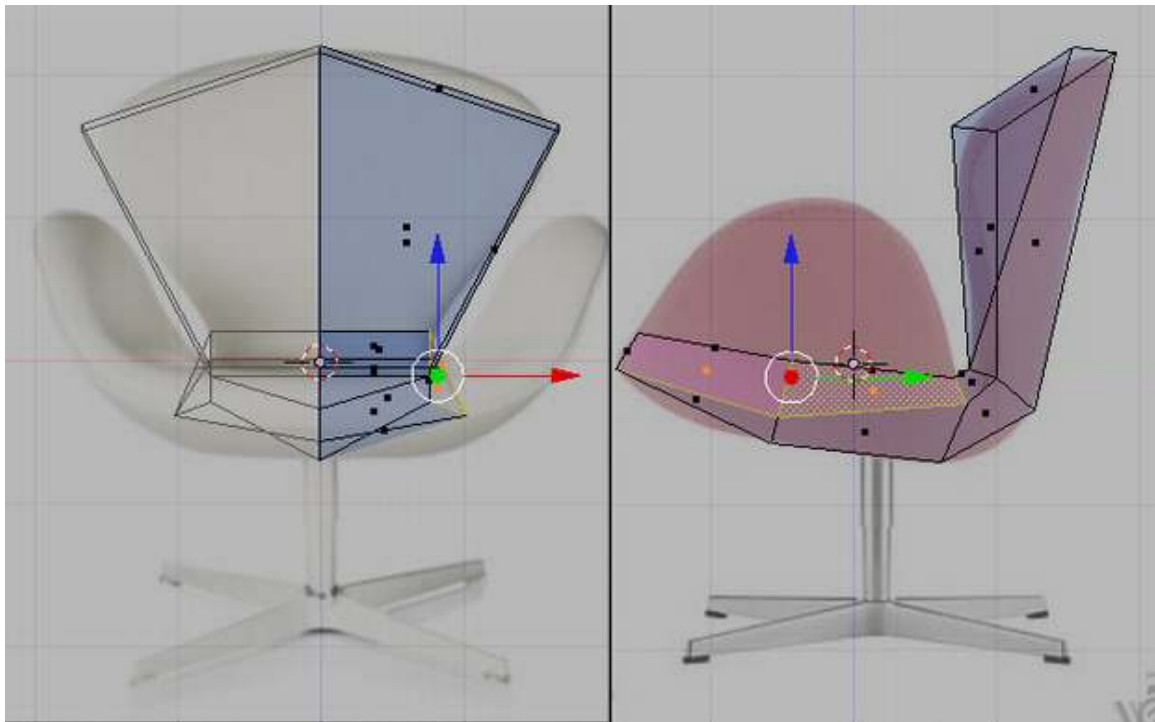


Figure 1465

Extrude the faces and adjust newly formed vertices to conform with the guide images.

p Shaping the sides

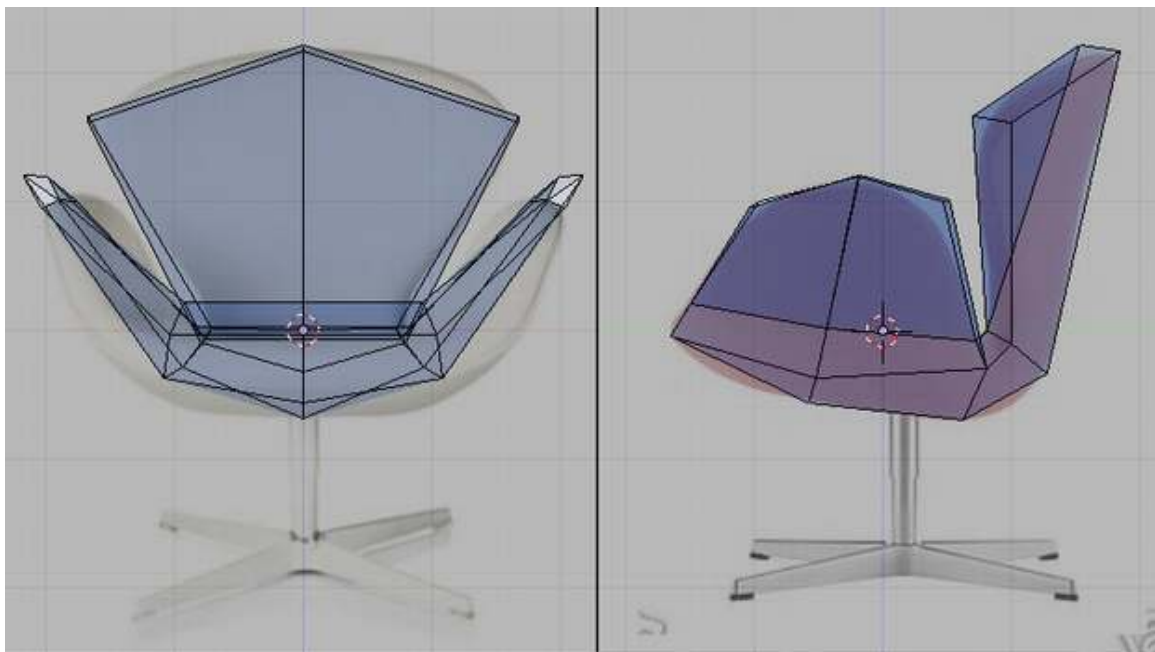


Figure 1466

And now the oblique view.

p Shaping the sides, oblique view

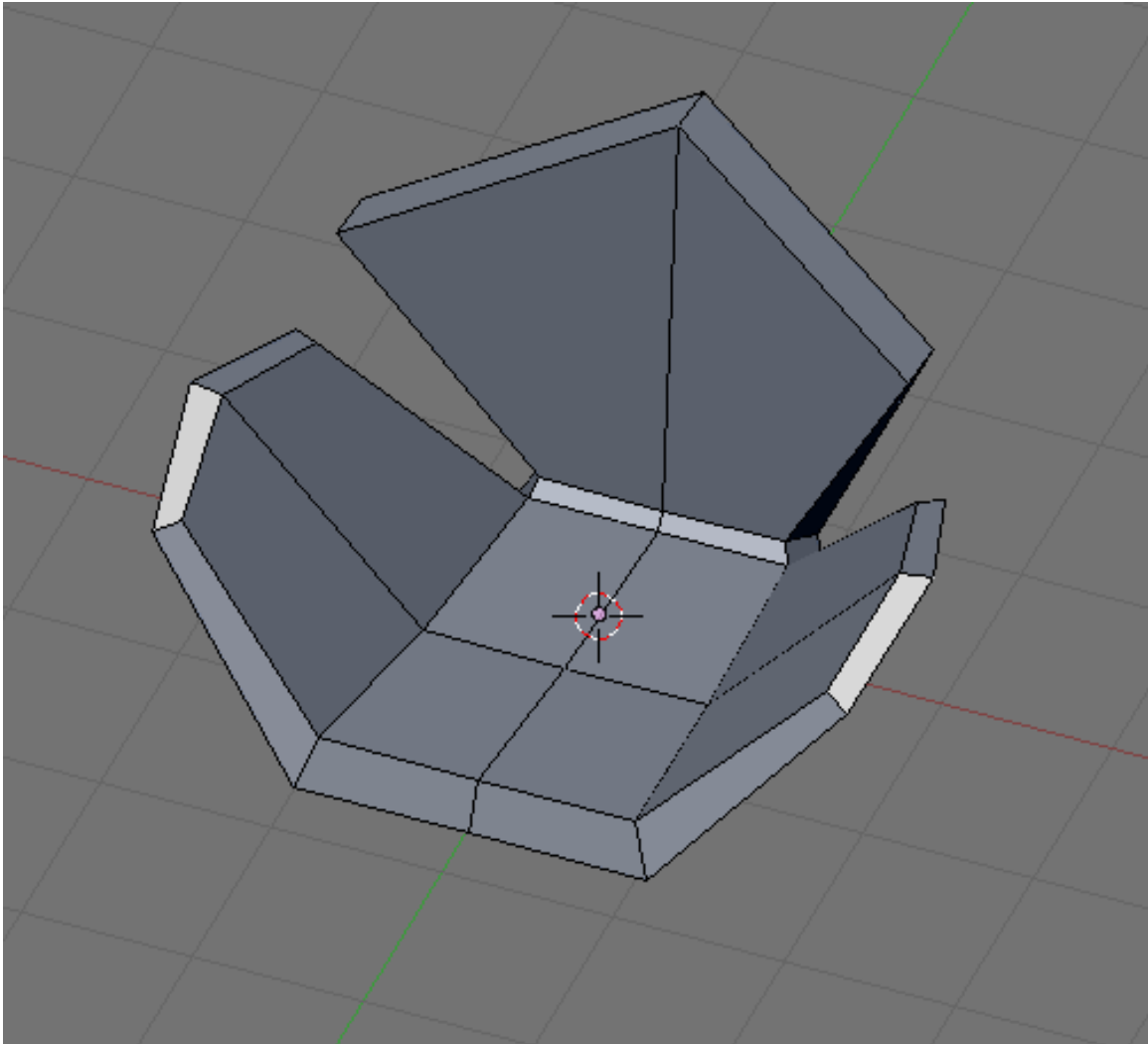


Figure 1467

The back still does not look good so add the following edge loop.

p Shaping the back again

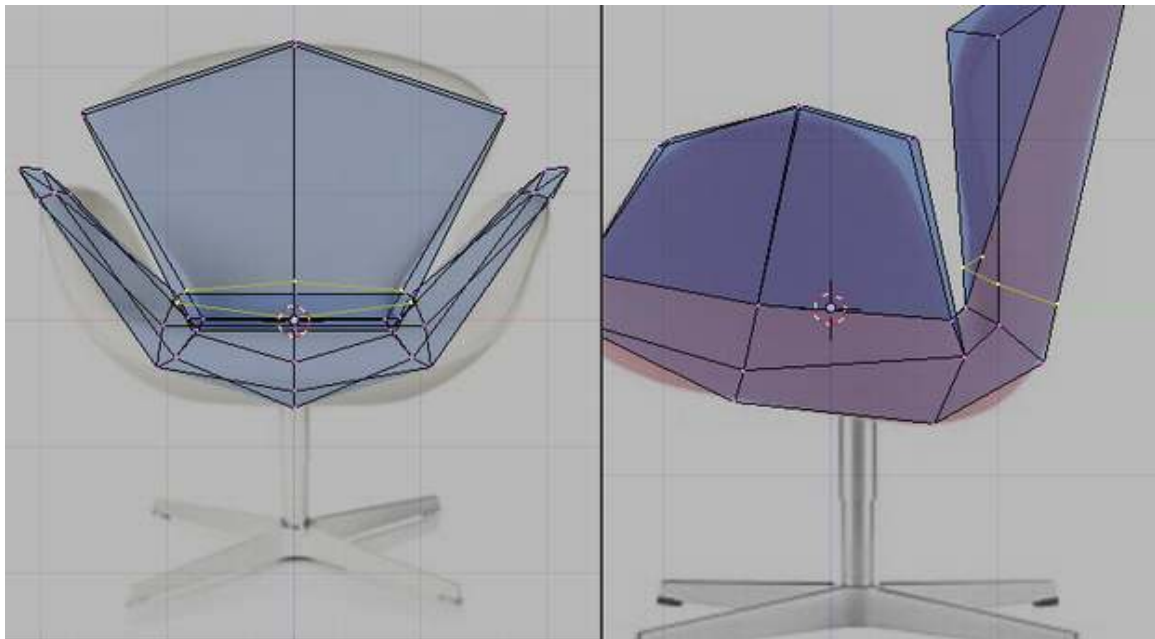


Figure 1468

Adjust it.

p Shaping the back again

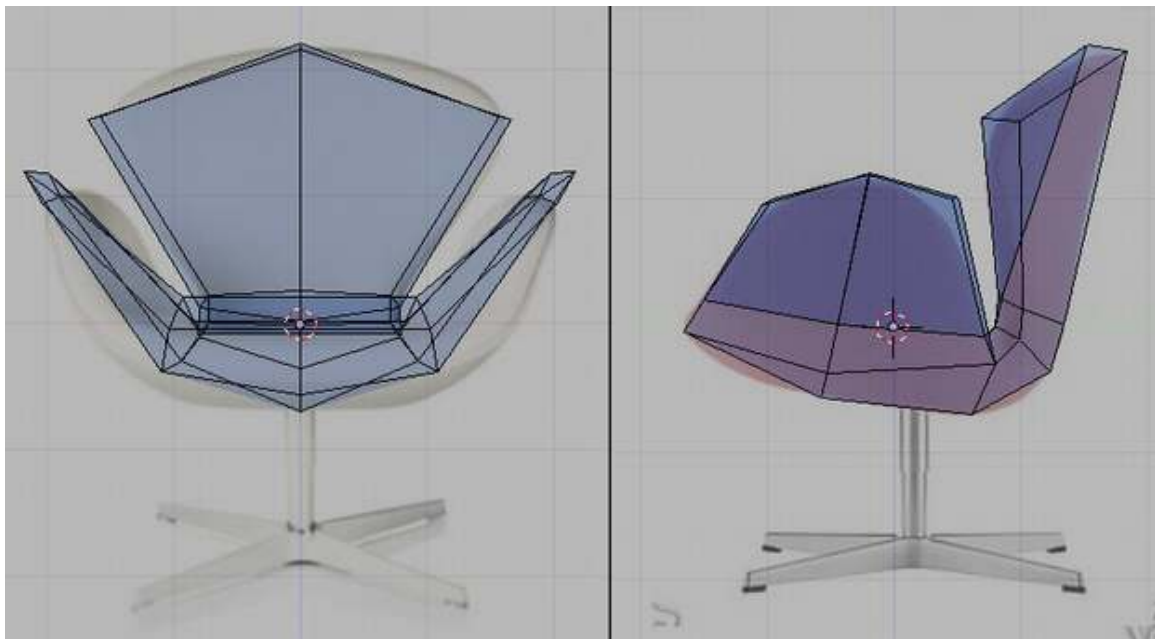


Figure 1469

Remember to rotate your view to see problem areas.

p Shaping the back again, oblique view

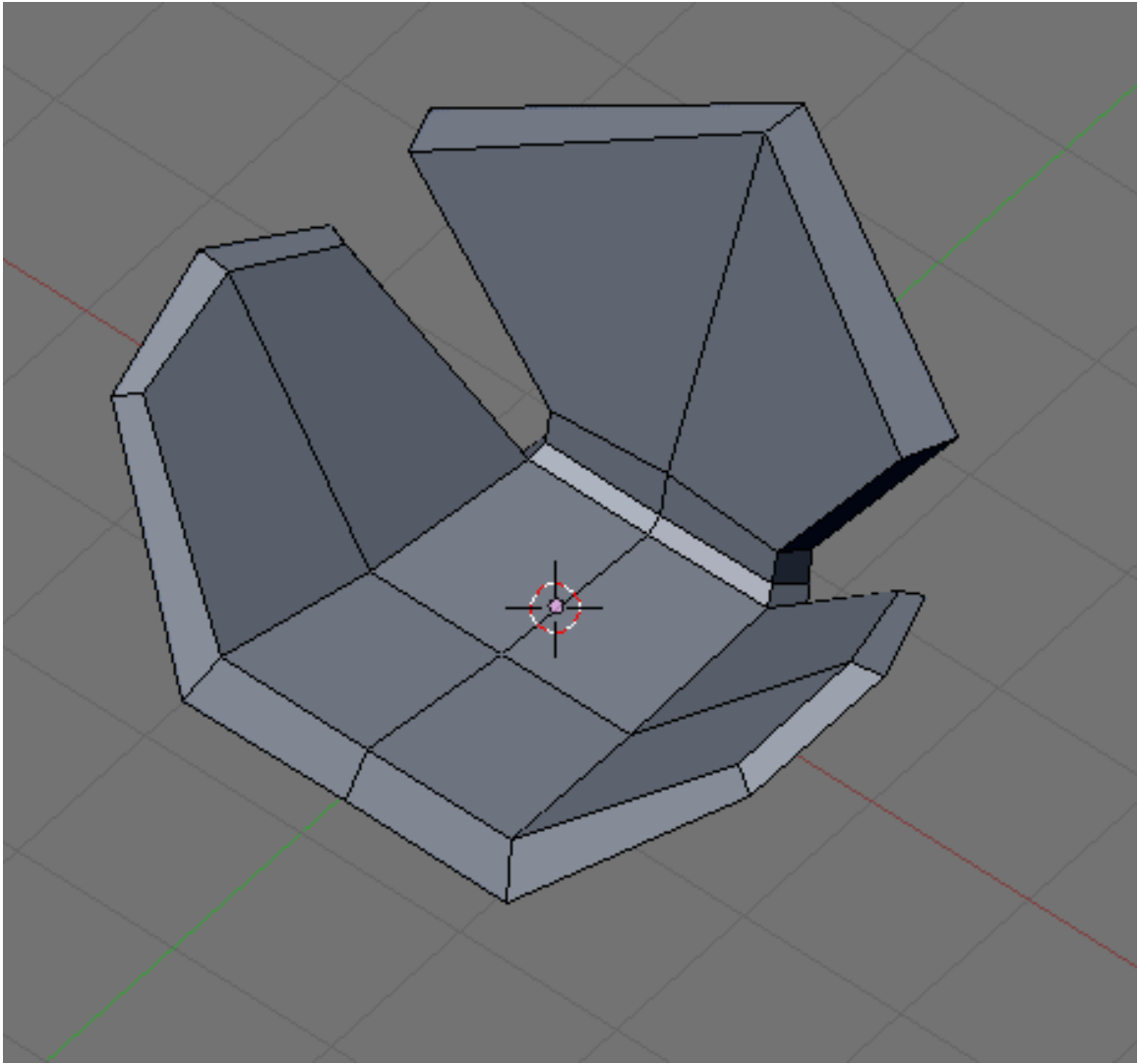


Figure 1470

Add this edge loop too.

p Shaping the back again

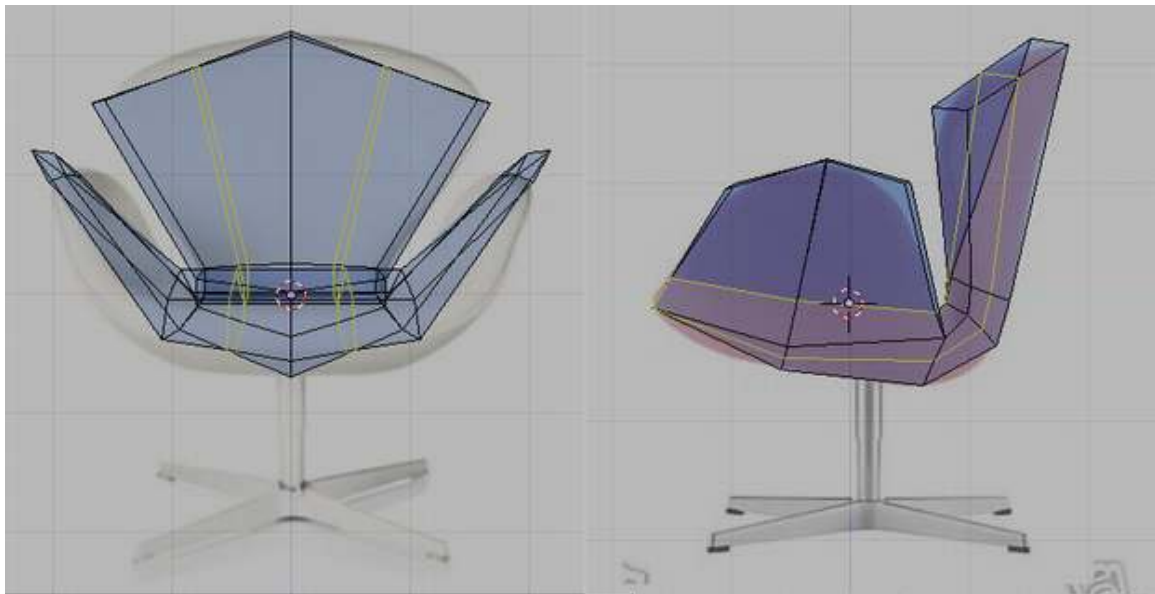


Figure 1471

Adjust.

p Shaping the back again

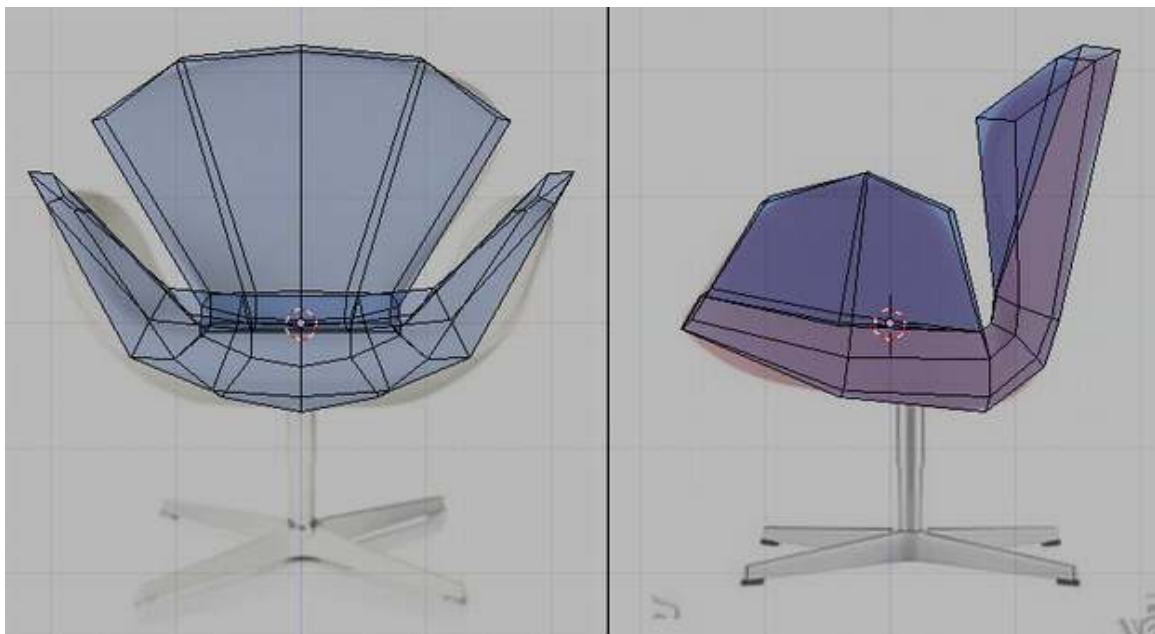


Figure 1472

And keeping with our good practice.

p Shaping the back again, oblique view

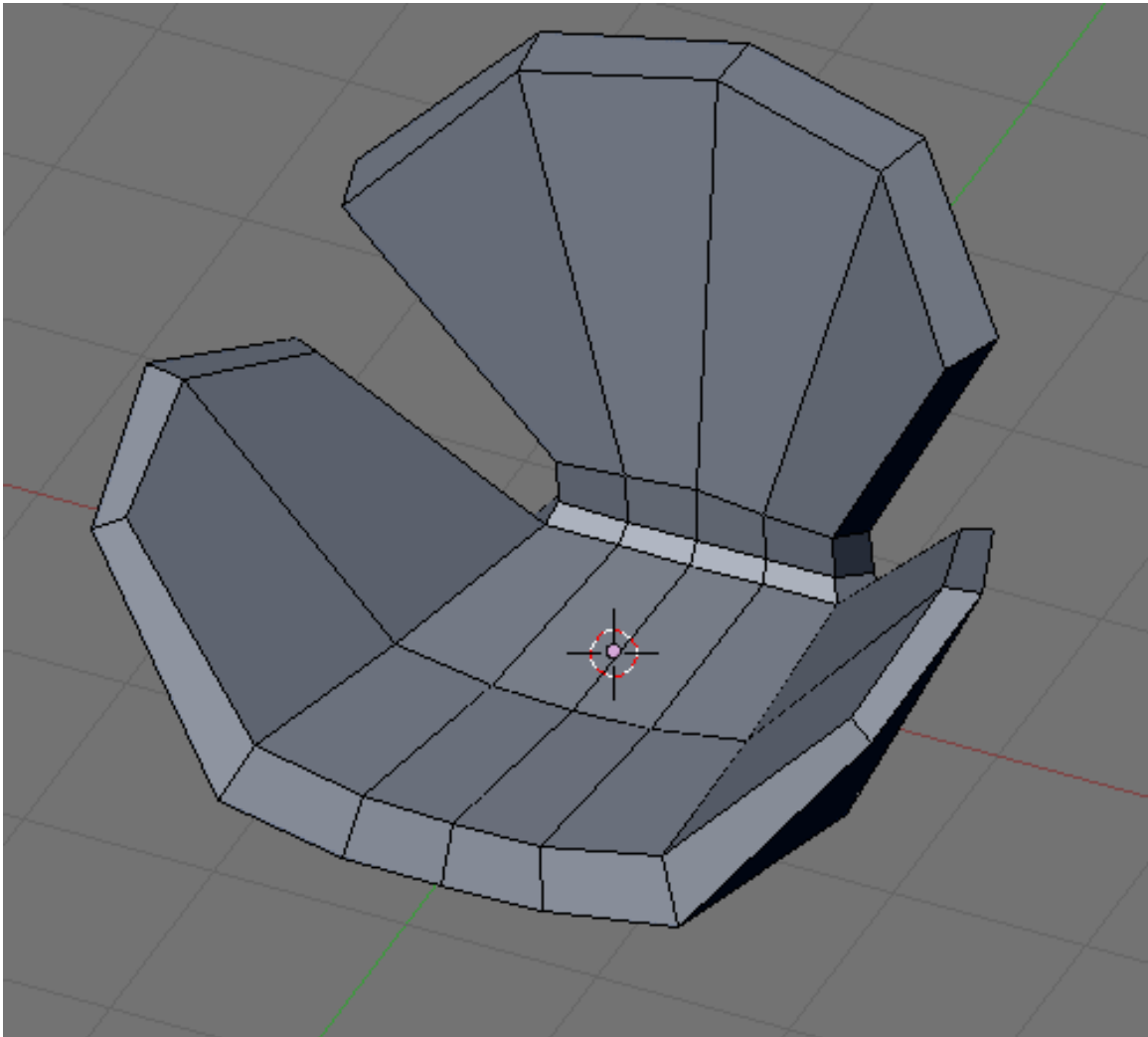


Figure 1473

Say, the form is finished. The loops are all in place and the form is easily identified as a rough model of the Swan chair.

232.3. Adding More Details

Now the basic form of the model is finished and with all important loops in place detailing this model is easy. Lets start by defining the back. Add these edge loops.

p Add these loop

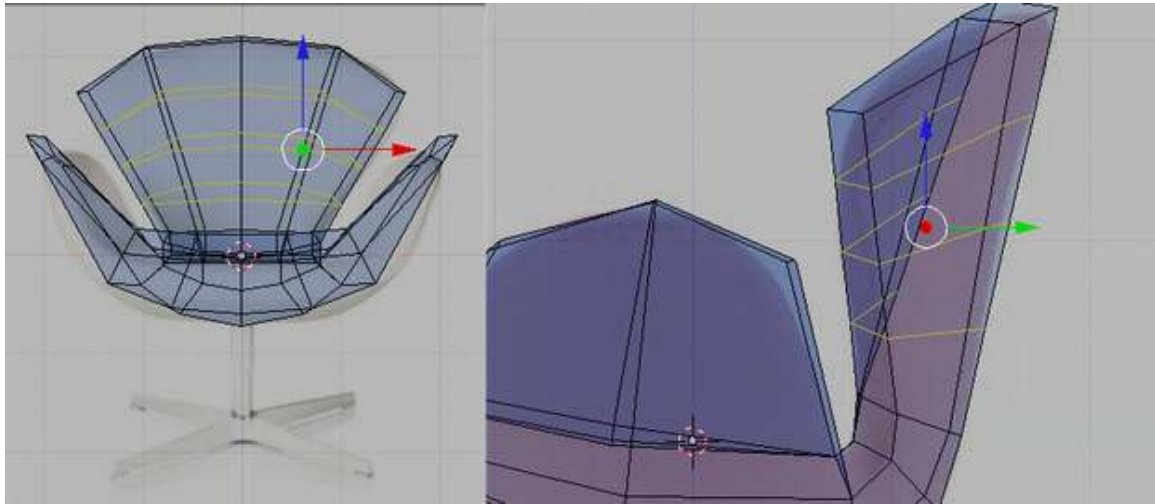


Figure 1474

Move the newly created vertices to fit the references.

p Adjust the form

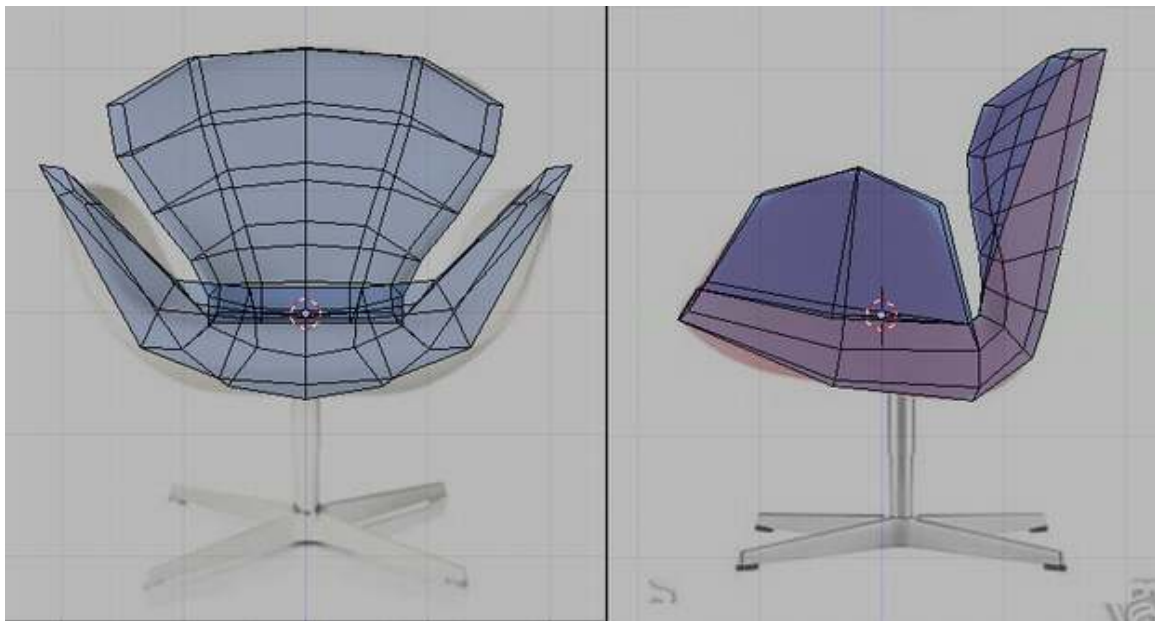


Figure 1475

Be sure that the model looks right at various angles. The references images collected earlier will help. Adjust vertices if necessary.

p Our model

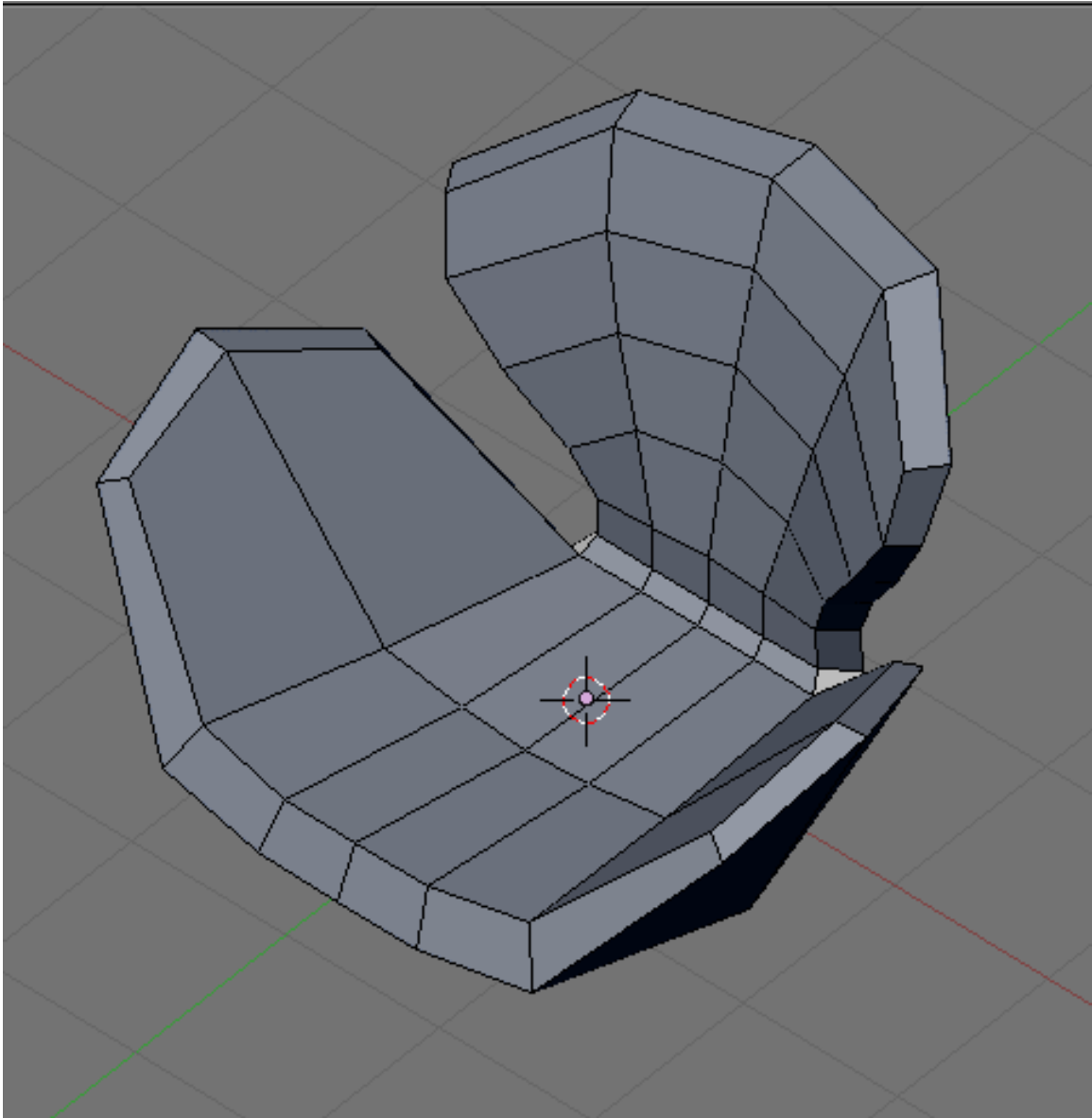


Figure 1476

The sides need to be taken care of too so add these edge loops.

p Add these loops

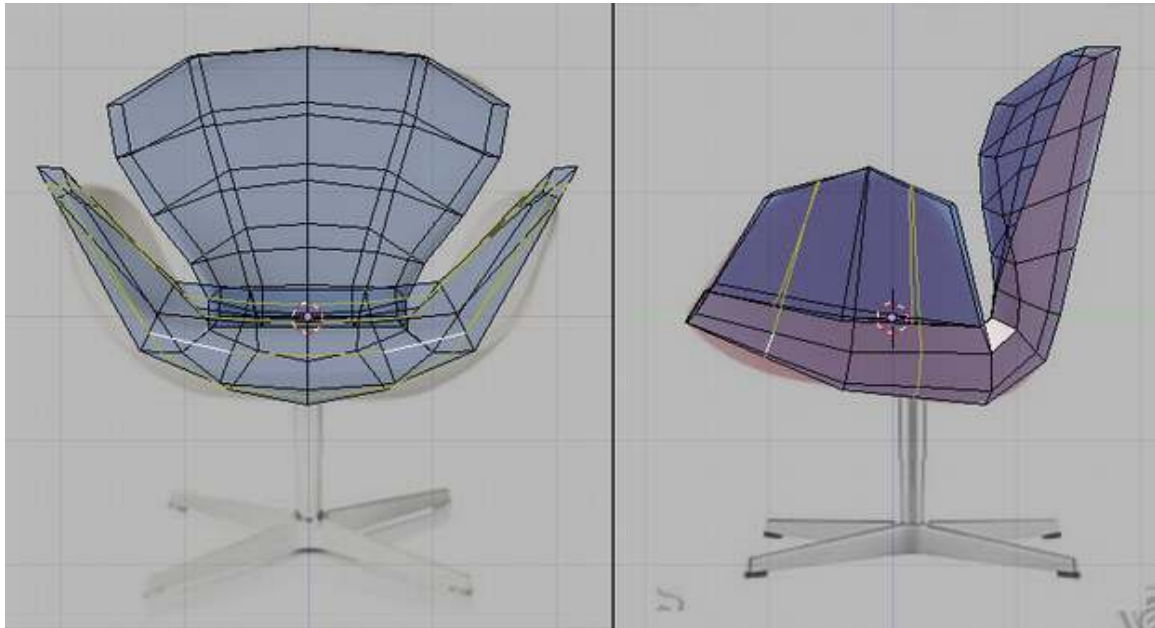


Figure 1477

Then adjust the vertices.

p Refine

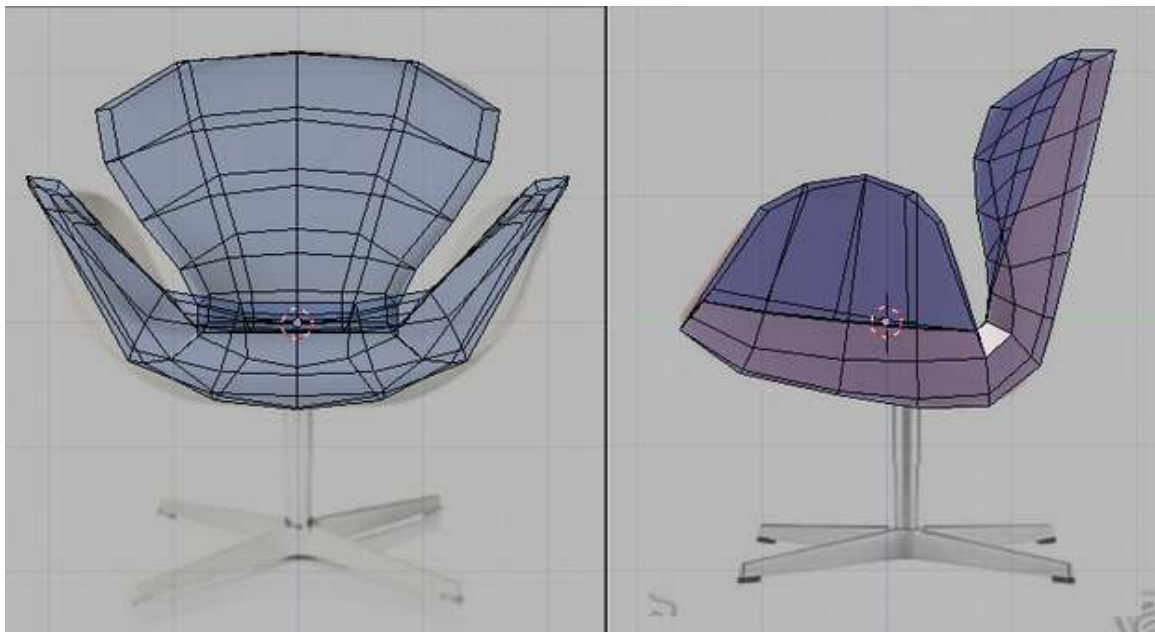


Figure 1478

This is how the model now looks like.

p Our model

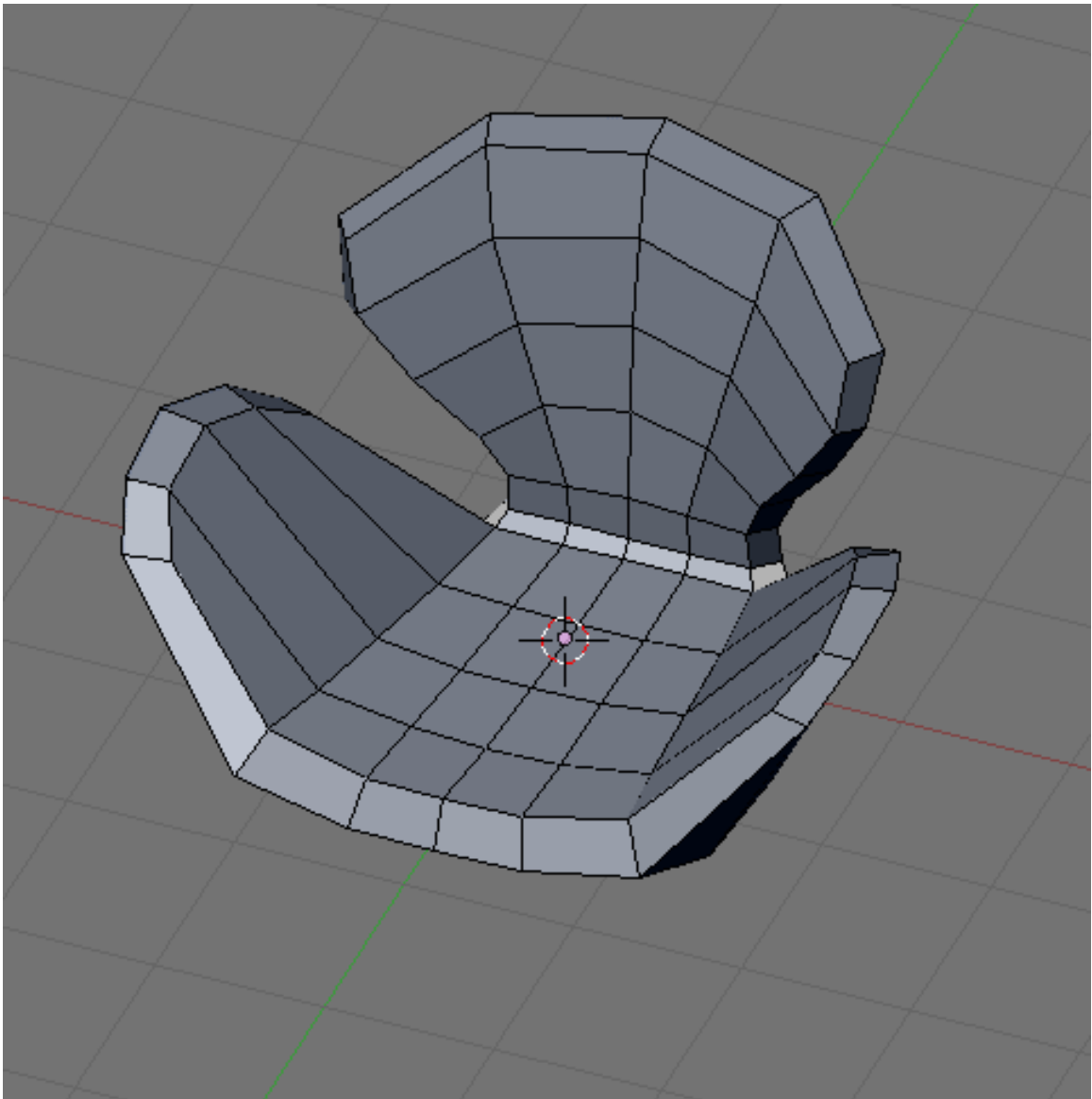


Figure 1479

Continue by adding these loops.

p Add these loops

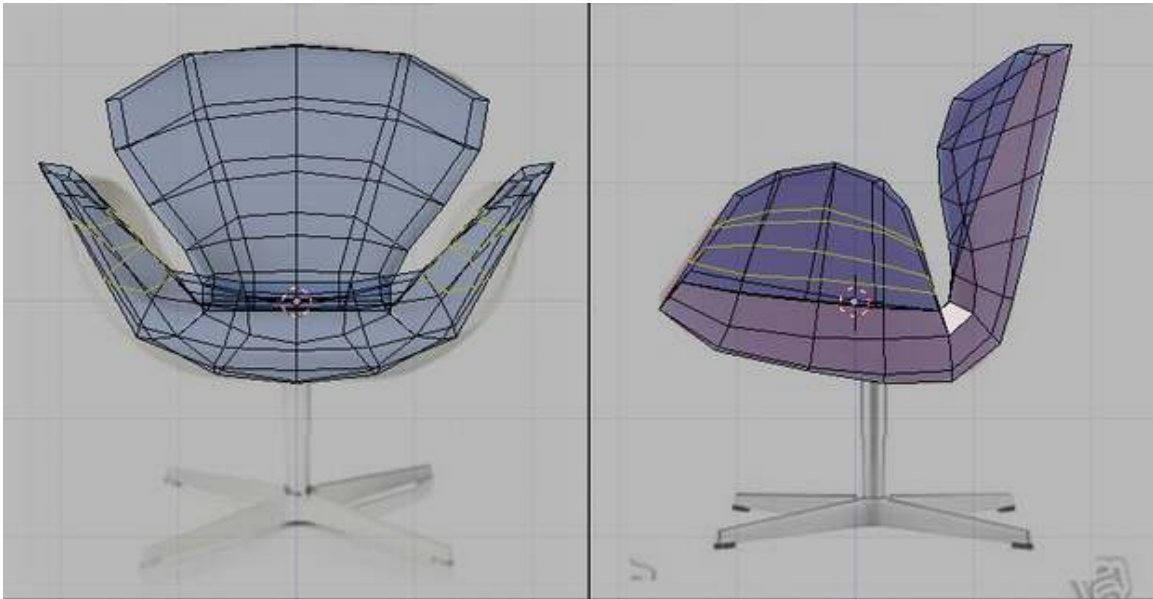


Figure 1480

And adjust as usual.

p Tweak

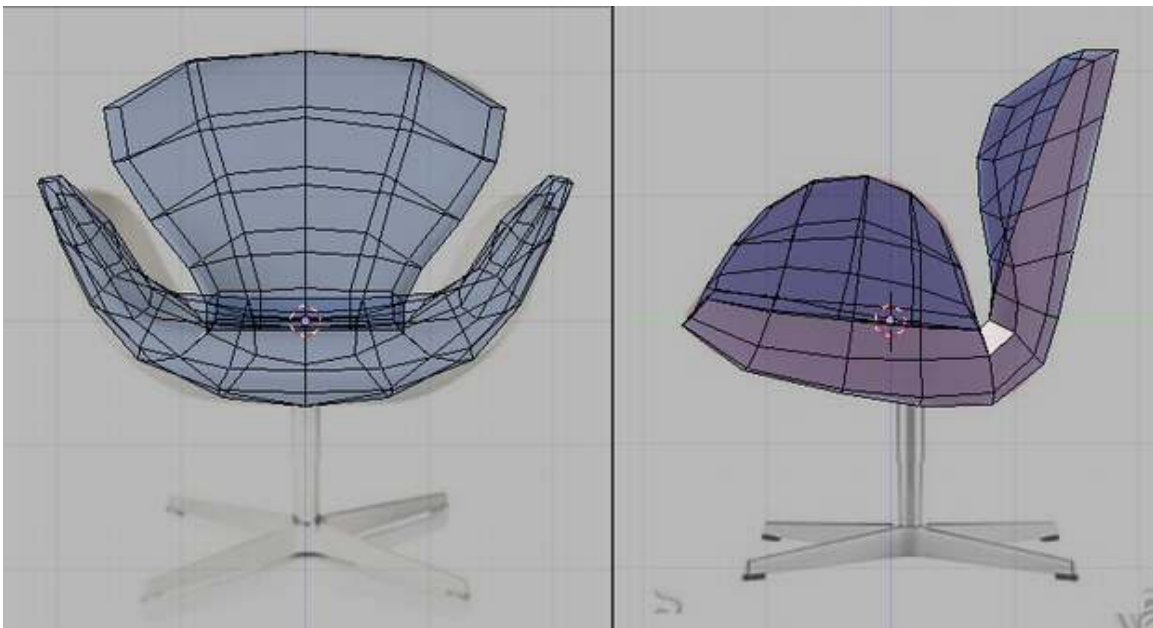


Figure 1481

Check that it looks good at other views too. Move vertices as necessary.

p Our model

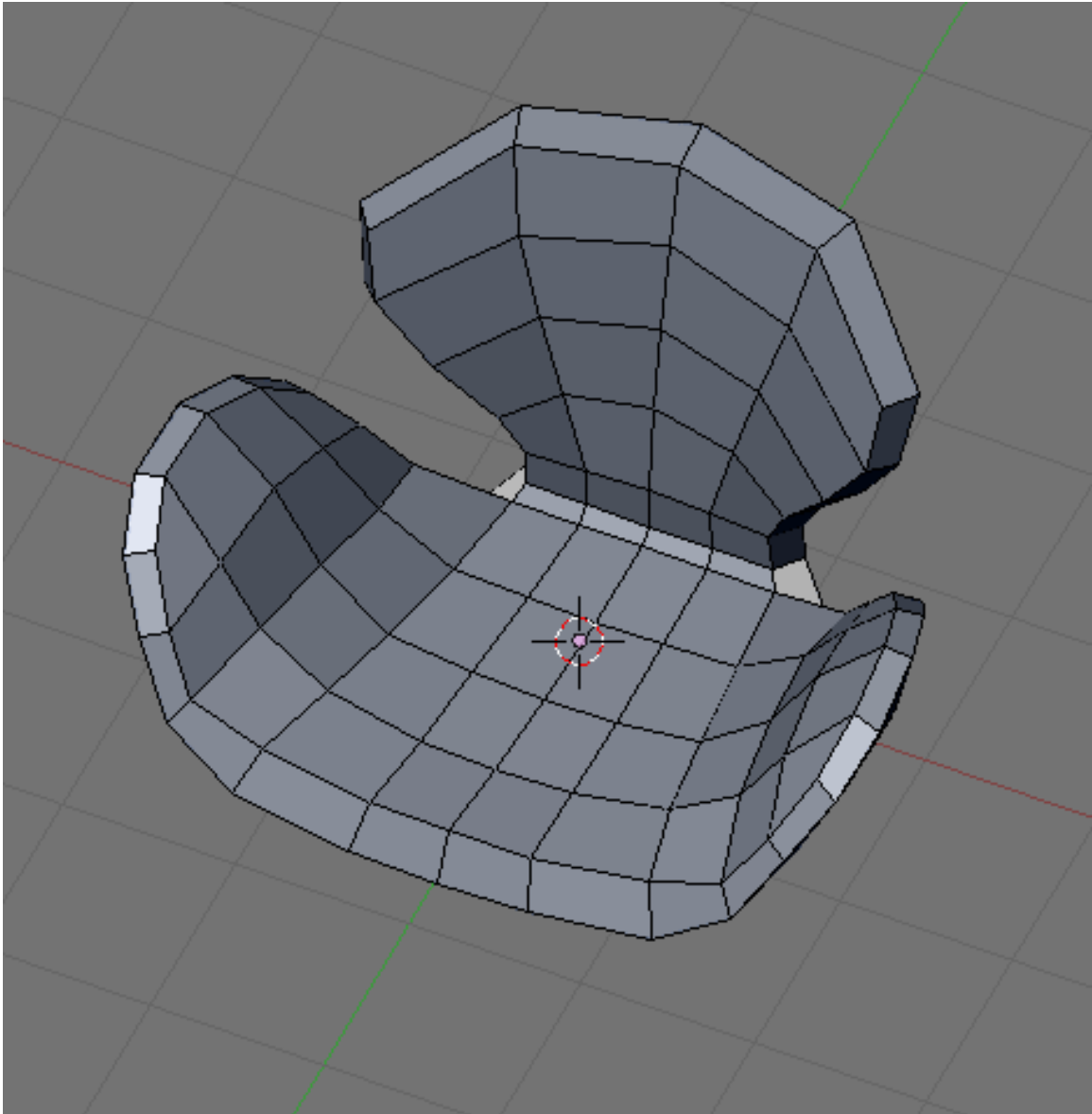


Figure 1482

Keep repeating this procedure. Add loops then adjust. Here is the loops that was added to the model. Tweaked after adding each.

p More loops

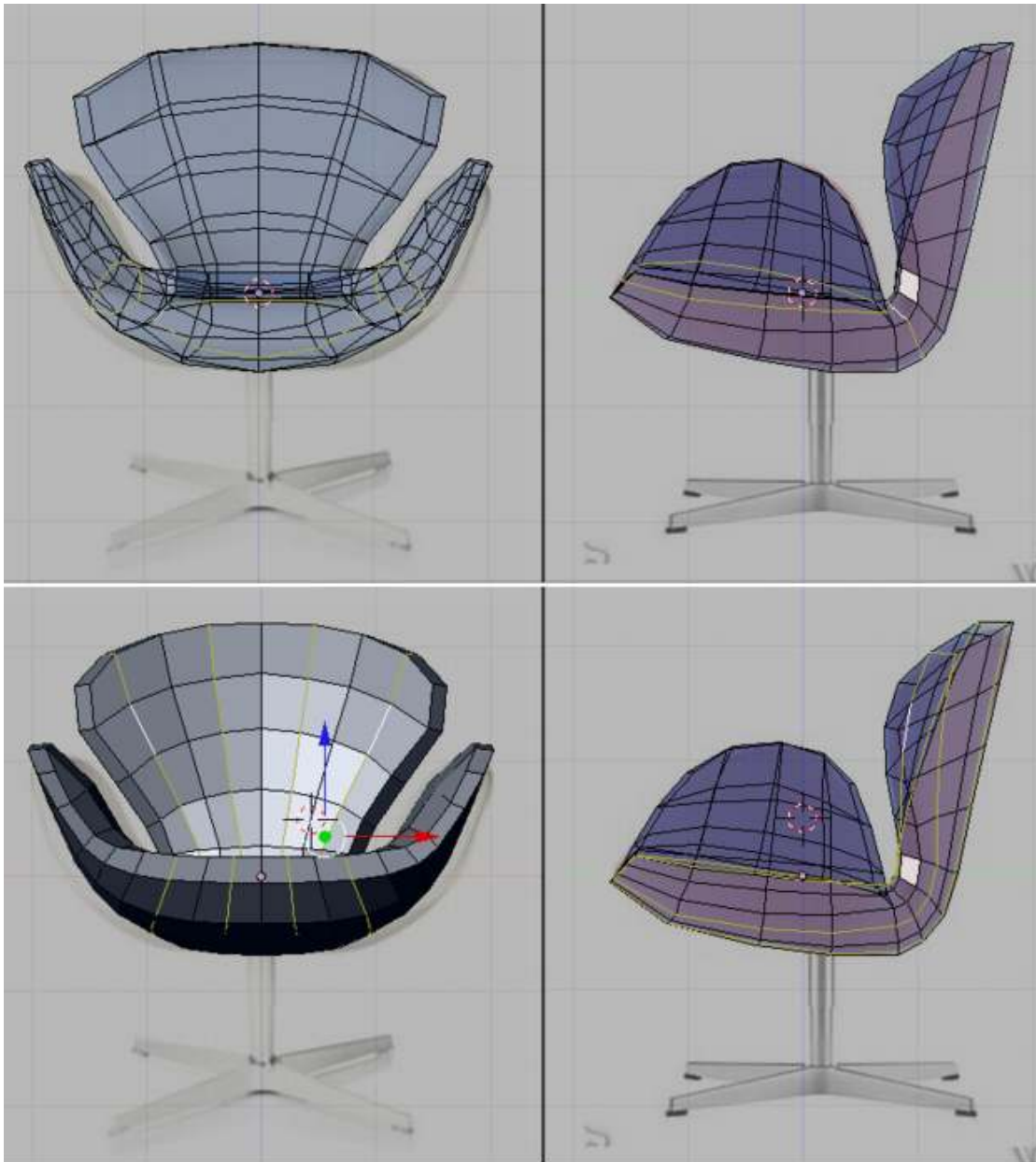


Figure 1483

The outline of the chair needs rounding up. Add these loops and tweak so it looks like the one shown.
p Rounding loops

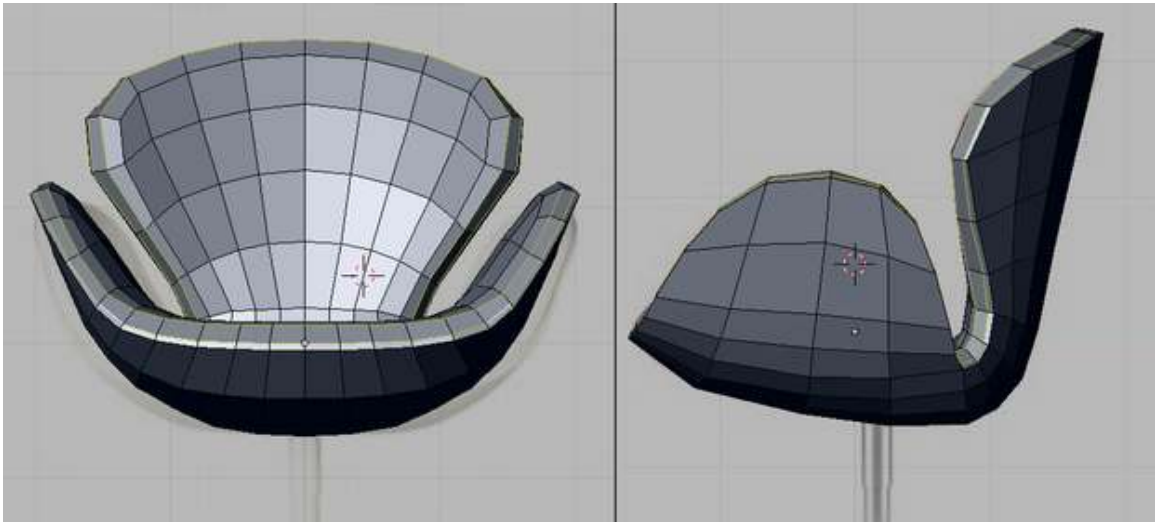


Figure 1484

This is now how the mode looks like.

p Our model

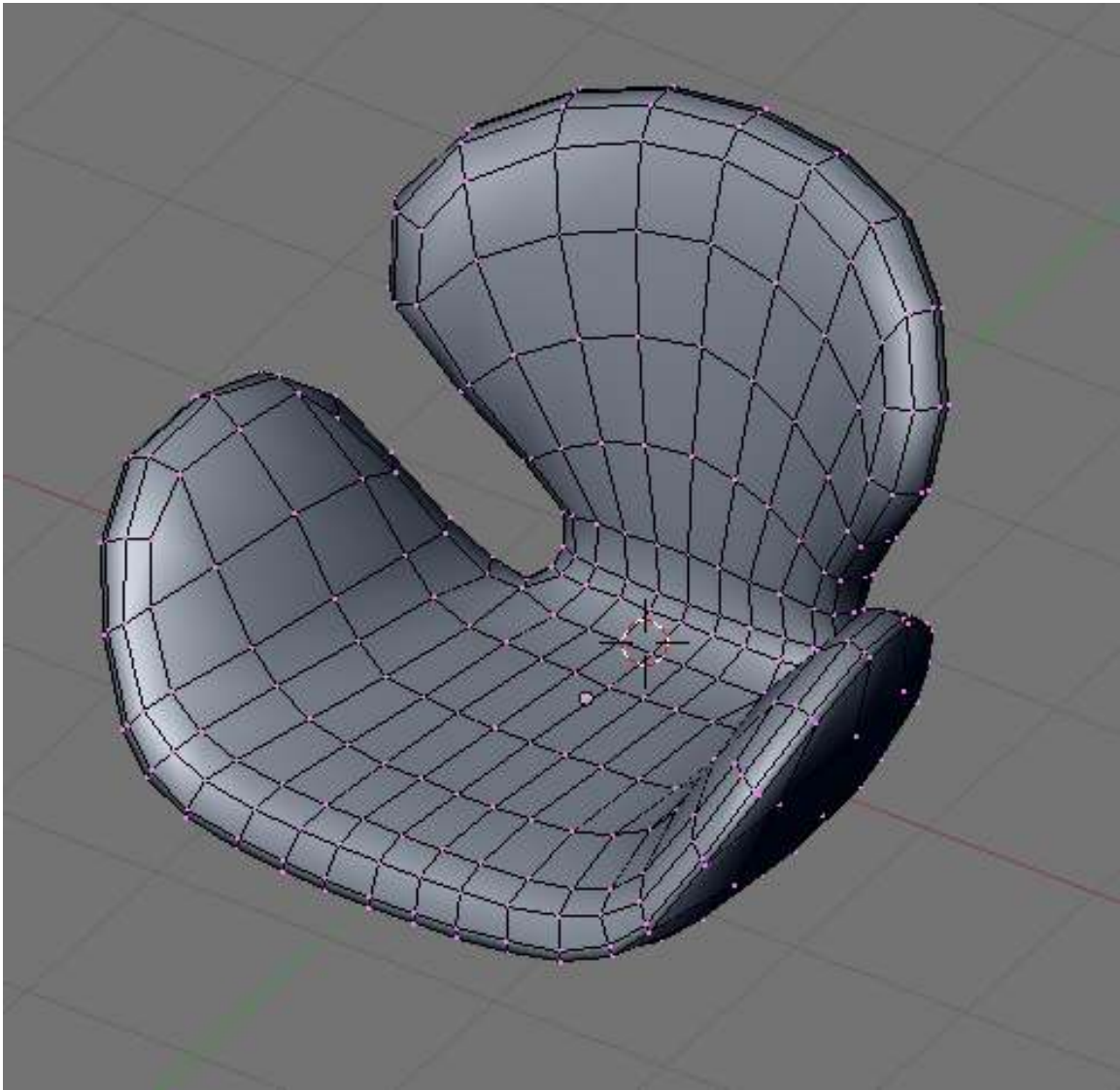


Figure 1485

Its rather tedious to add more loops at this time so lets do a cheat. Select all vertices (press AKey a few times) and then subdivide (WKey -> IKey) to add more vertices.

p Subdivide the whole mesh

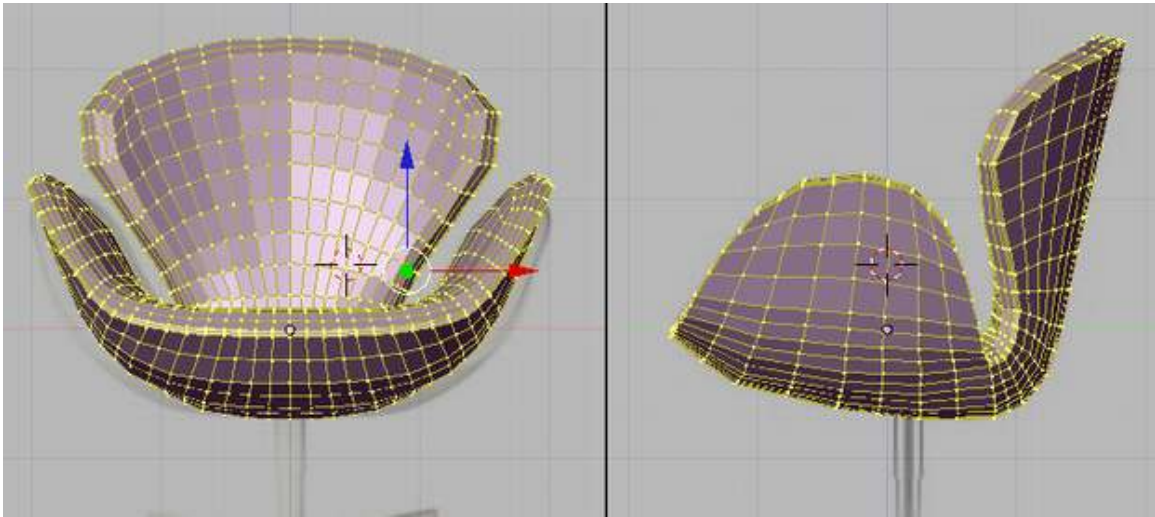


Figure 1486

The model looks blocky so lets smooth it out. While the whole chair is still selected smooth the vertices by clicking Smooth a few times (Edit buttons ->Mesh tools -> Smooth; also available at WKey->Smooth). It would smooth out the model.

p Smoothing the model

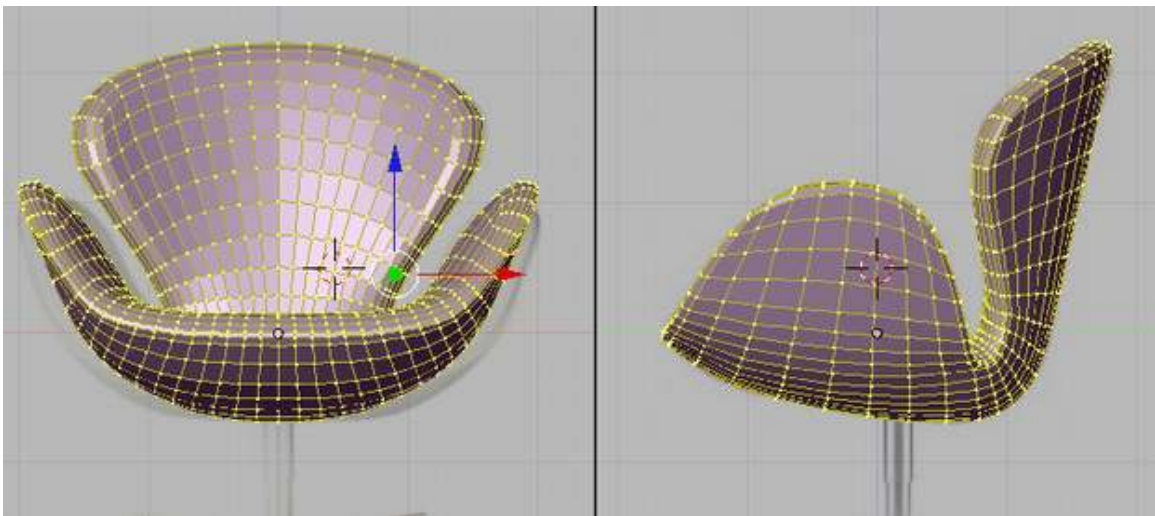


Figure 1487

Admire the work (and tweak if necessary).

p Our model

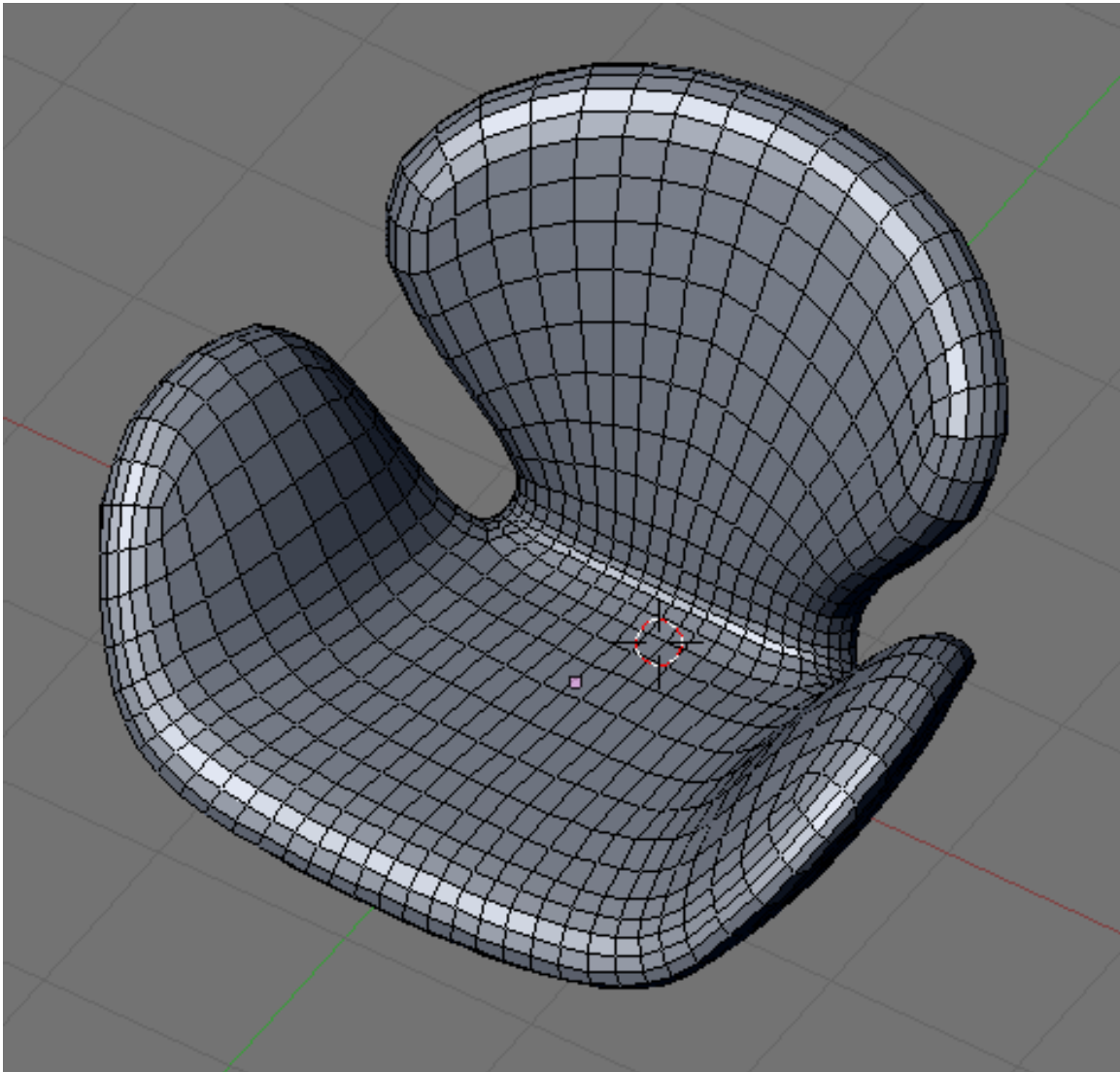


Figure 1488

This is optional. Add this loop and scale along normal (Alt + Skey) to scale it in. Adjust vertices if necessary.

p Modeling the seams

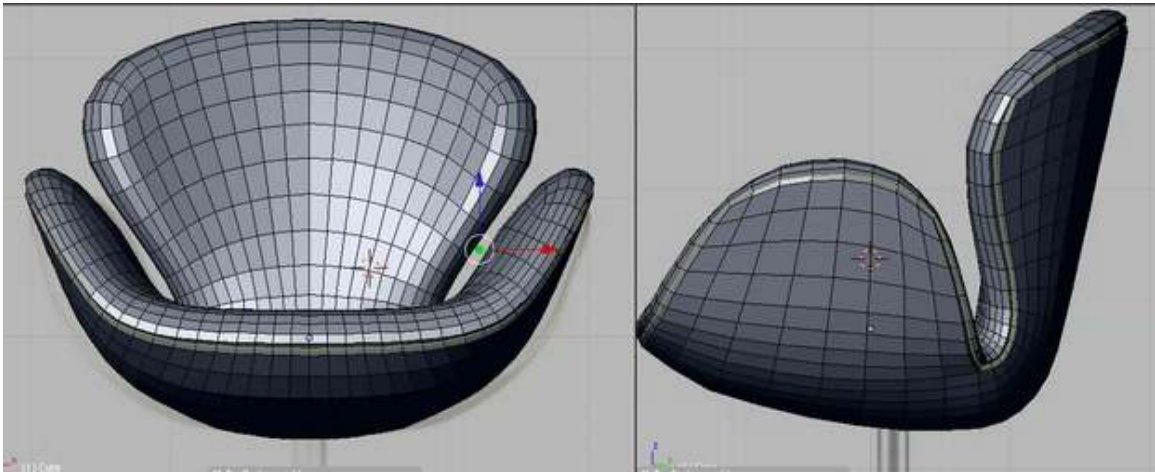


Figure 1489

233. Model a Chair-The Feet

Now the seat is finished, time for the legs for it to stand on.

233.1. Starting Primitive

The leg is mainly made up of cylinders so a cylinder is a very good primitive to start. In object mode add a cylinder with 16 vertices.Center the cylinder on the upper part of the telescope.

p Adding the cylinder primitive

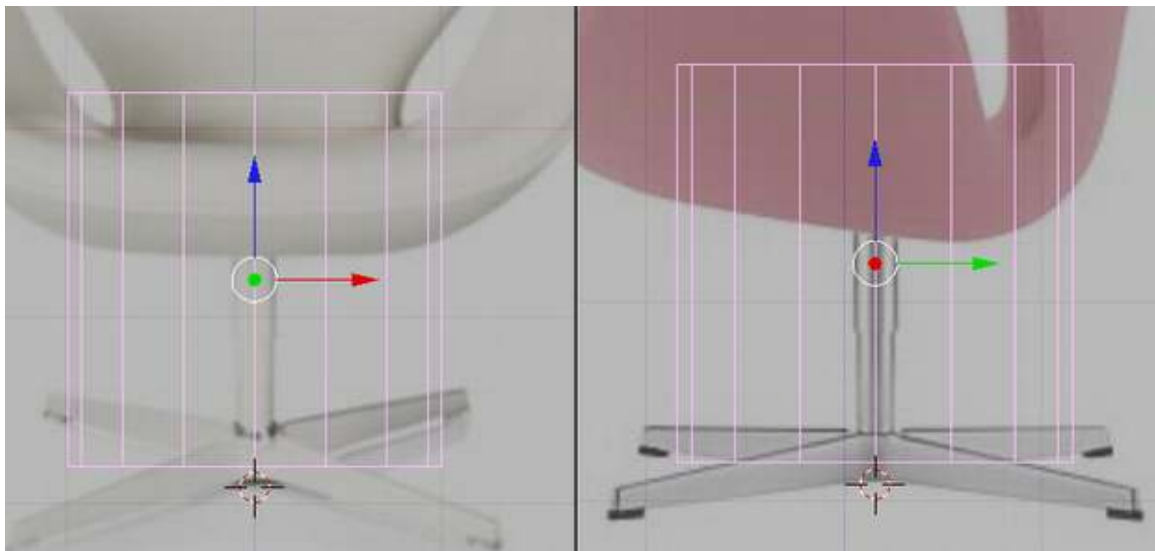


Figure 1490

Tip: to make the cylinder centered relative to the seats center, while in object mode select the seat then hit Shift+Skey (Snap options) then select Cursor->Selection. The 3d cursor would be moved to the center of our seat. Now when you add the cylinder, the cylinders center would be located at the cursor which is also at the seats center. Now all you have to do is to move the cylinder along the y and z axis because it would be properly located relative to the x axis.

Hide or move the Seat to another layer so that it would not intrude while modeling the leg.

233.2. The Upper Telescope

Go to edit mode and scale the cylinder until its radius match the radius of the upper telescopic element of the chair's leg. Then move the top and bottom vertices to match the reference.

p Modifying the primitive to fit reference

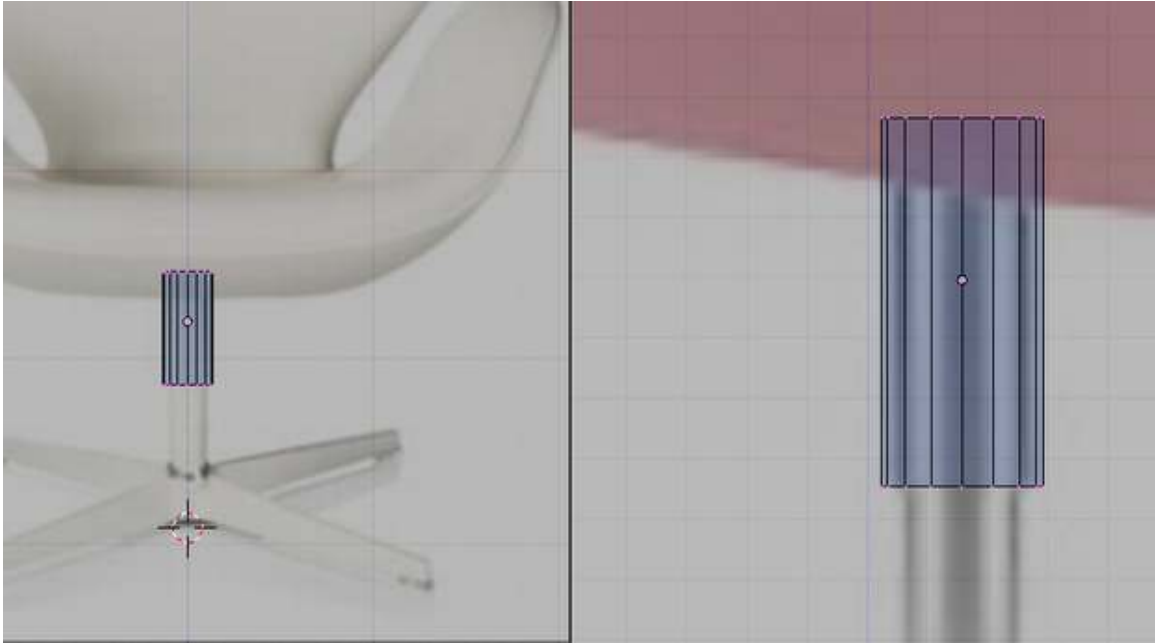


Figure 1491

Extrude the bottom part and place it a little lower. Extrude again and right click so that the newly created vertices would remain in their position. Scale the vertices a little. Extrude again and move the extruded region up the cylinder.

p Shaping the bottom of the telescope

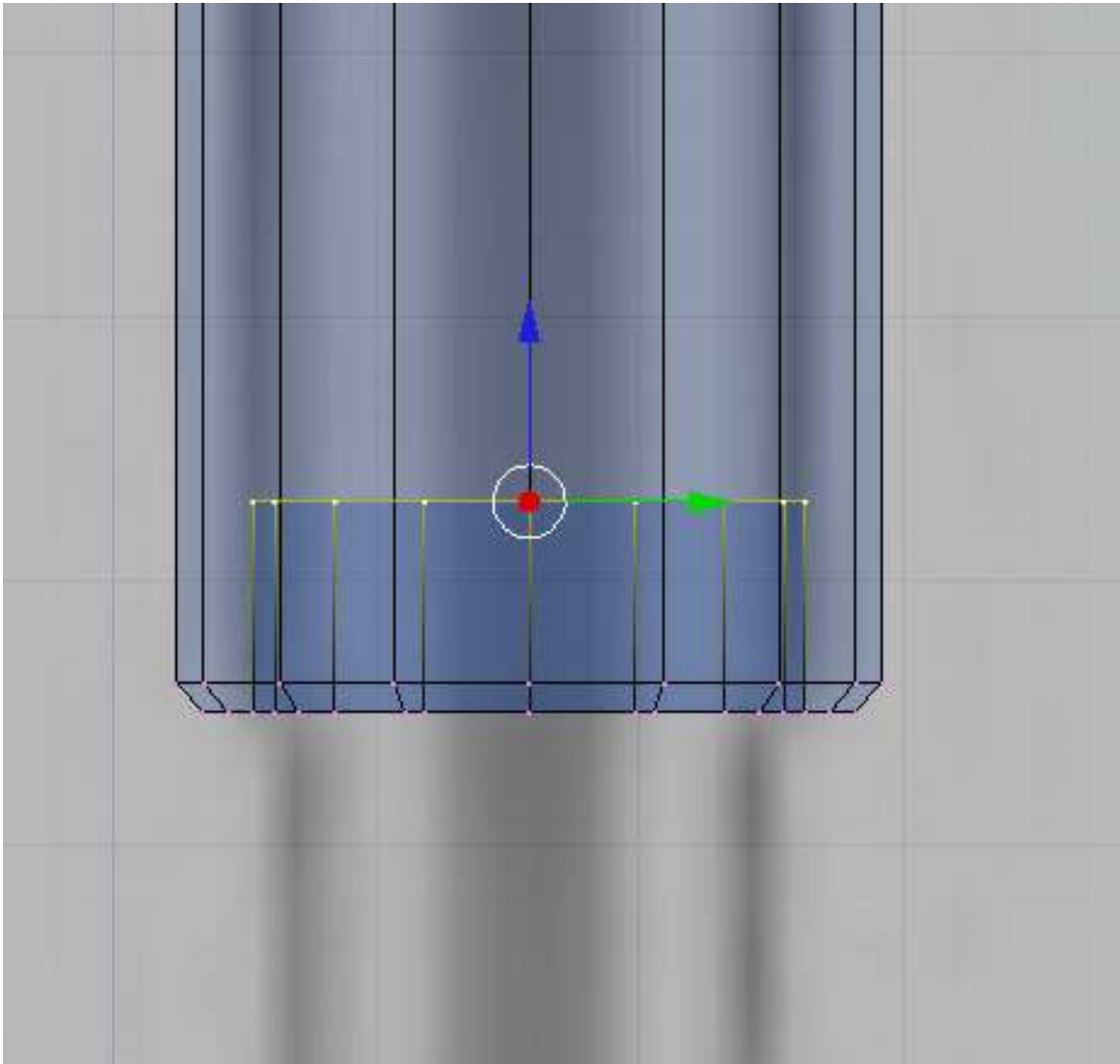


Figure 1492

Remove the top and bottom vertices that cap the ends.

p Opening ends

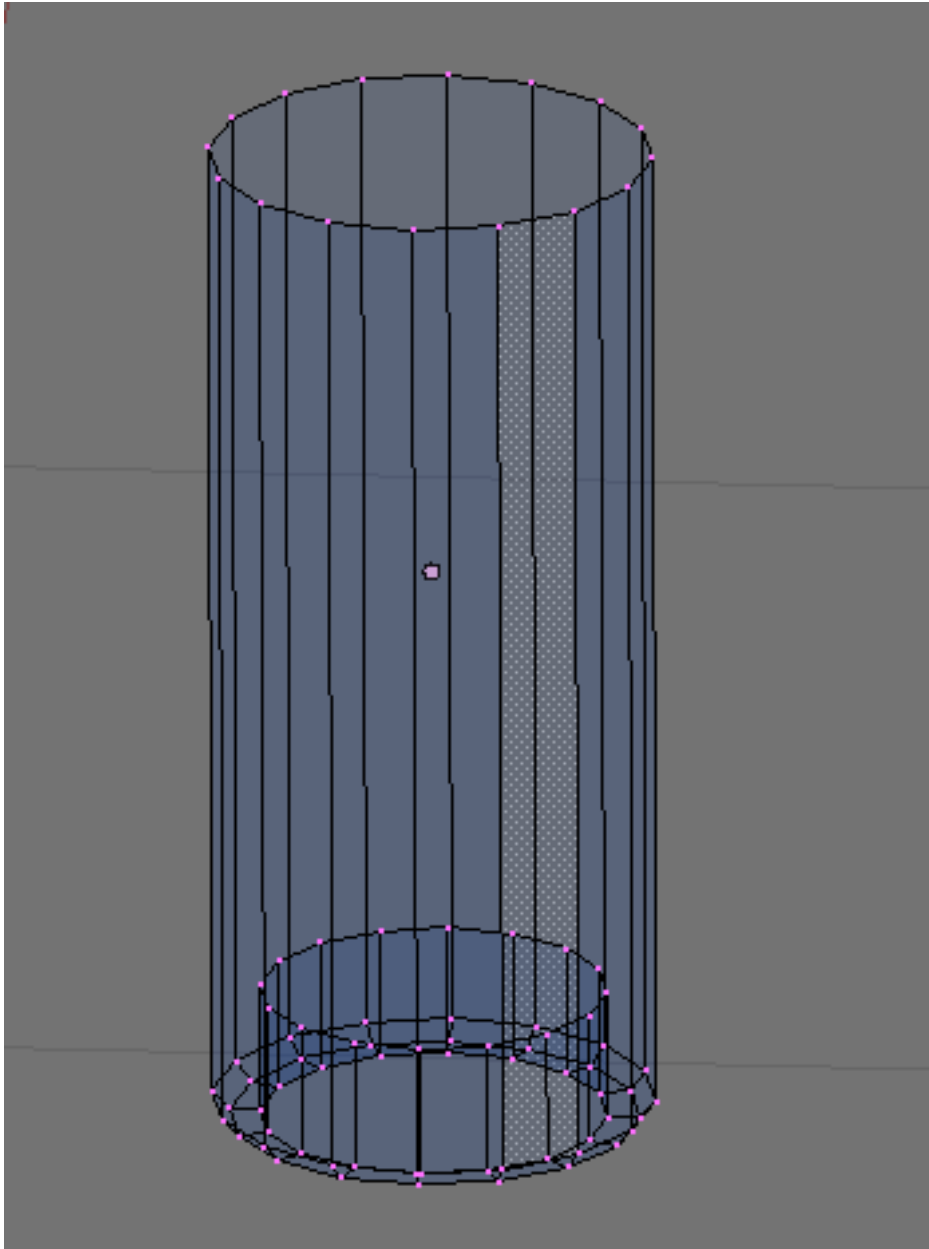


Figure 1493

We will be using subsurf modifier to make this part smoother. Add this loop to constrain the smoothing on this area.

p Adding control loop

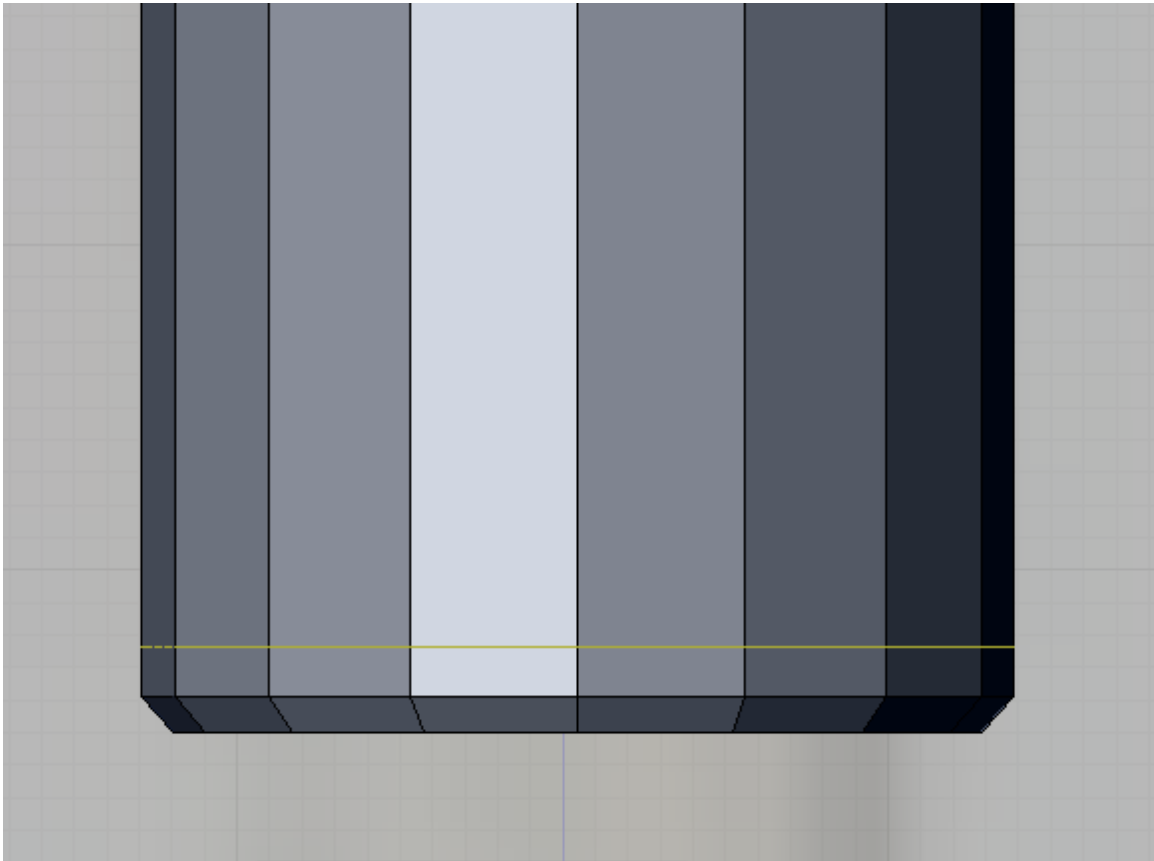


Figure 1494

233.3. The Lower Telescope

Now lets start with the lower part. Add a cylinder in object mode and adjust the newly created vertices to match the reference.

p Start modeling the lower telescopic part

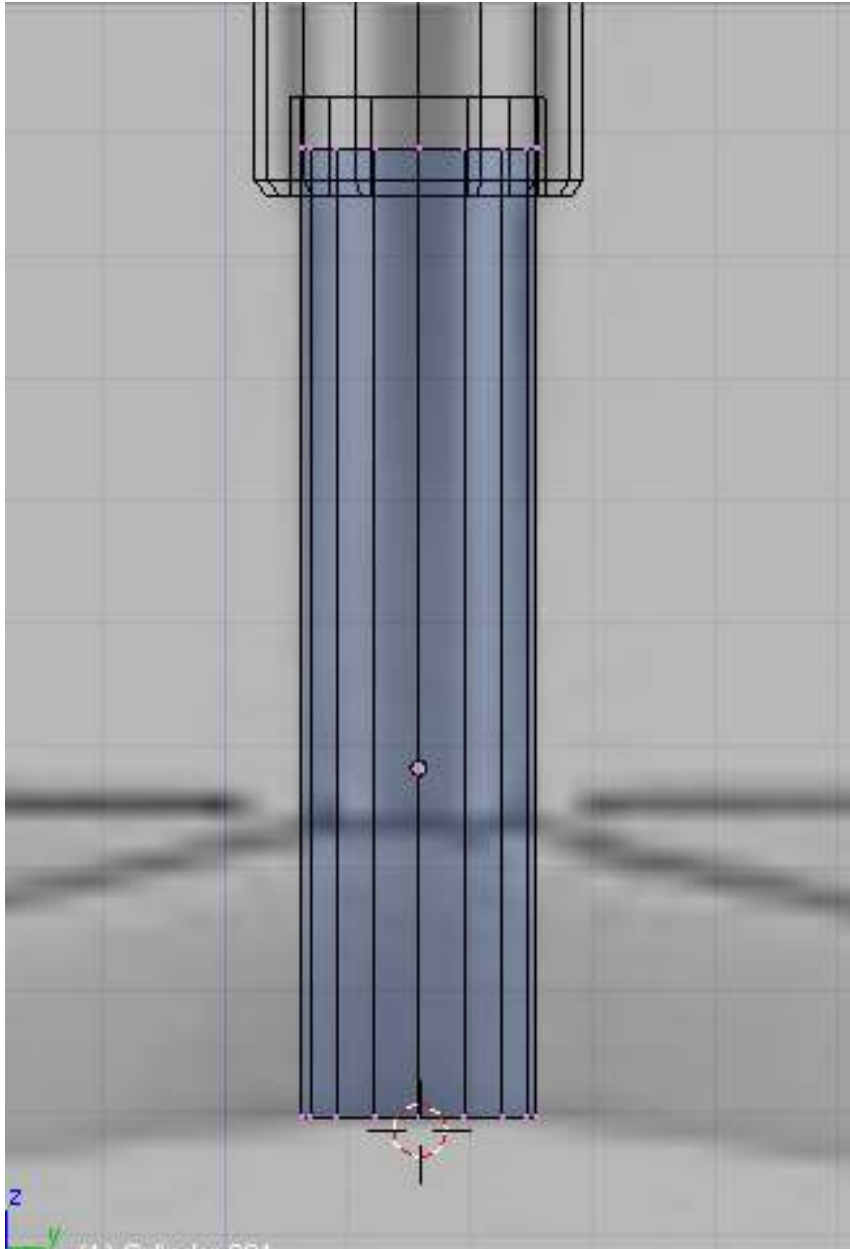


Figure 1495

The legs would be extruded from this cylinder. Add this loop to prepare.

p Add this loop to enable extrusion of legs

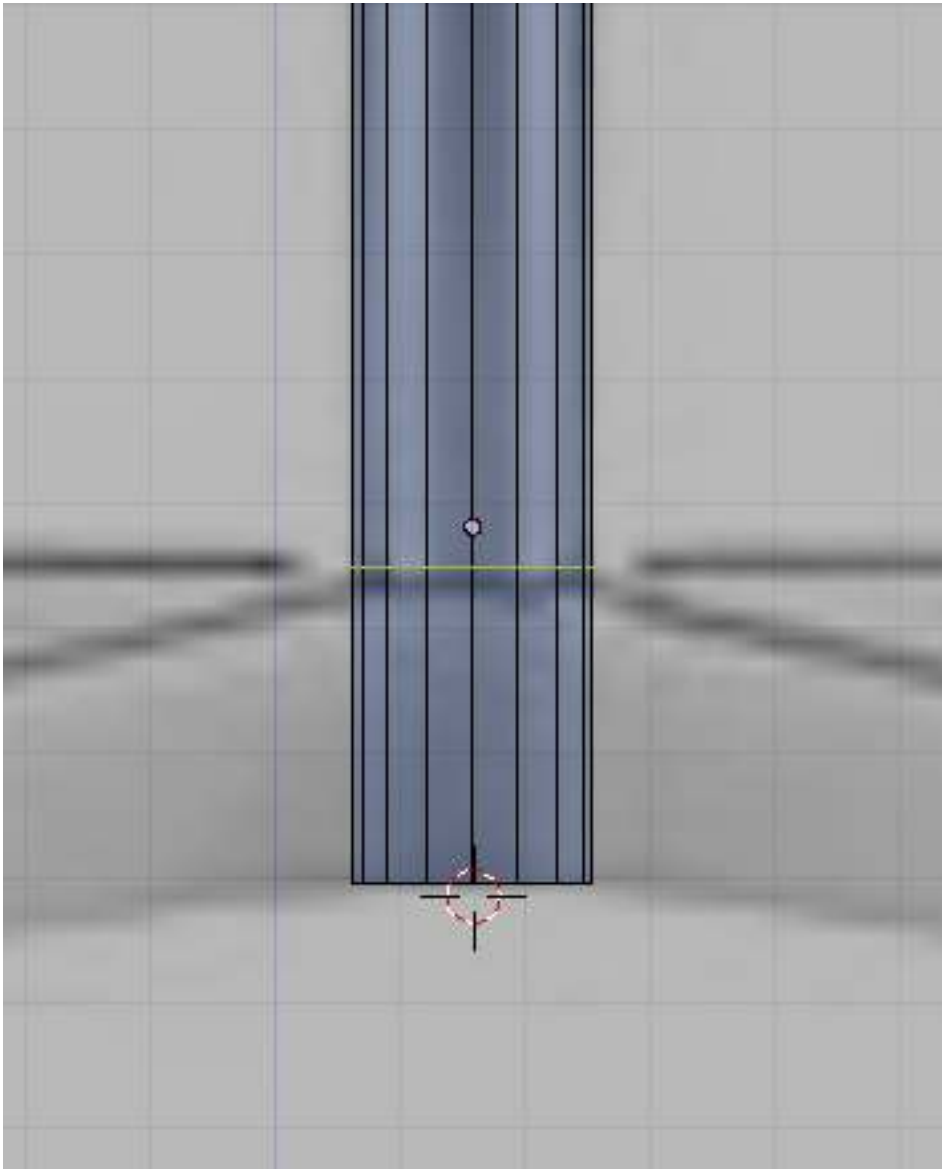


Figure 1496

Select the faces between the newly added loop and the bottom loop. Extrude this faces but right click after extrusion so the extruded faces remain in their position. Now scale them constrained along the xy-plane (Skey->Shift+Zkey) a little. This is how it should now look.

p Forming the base

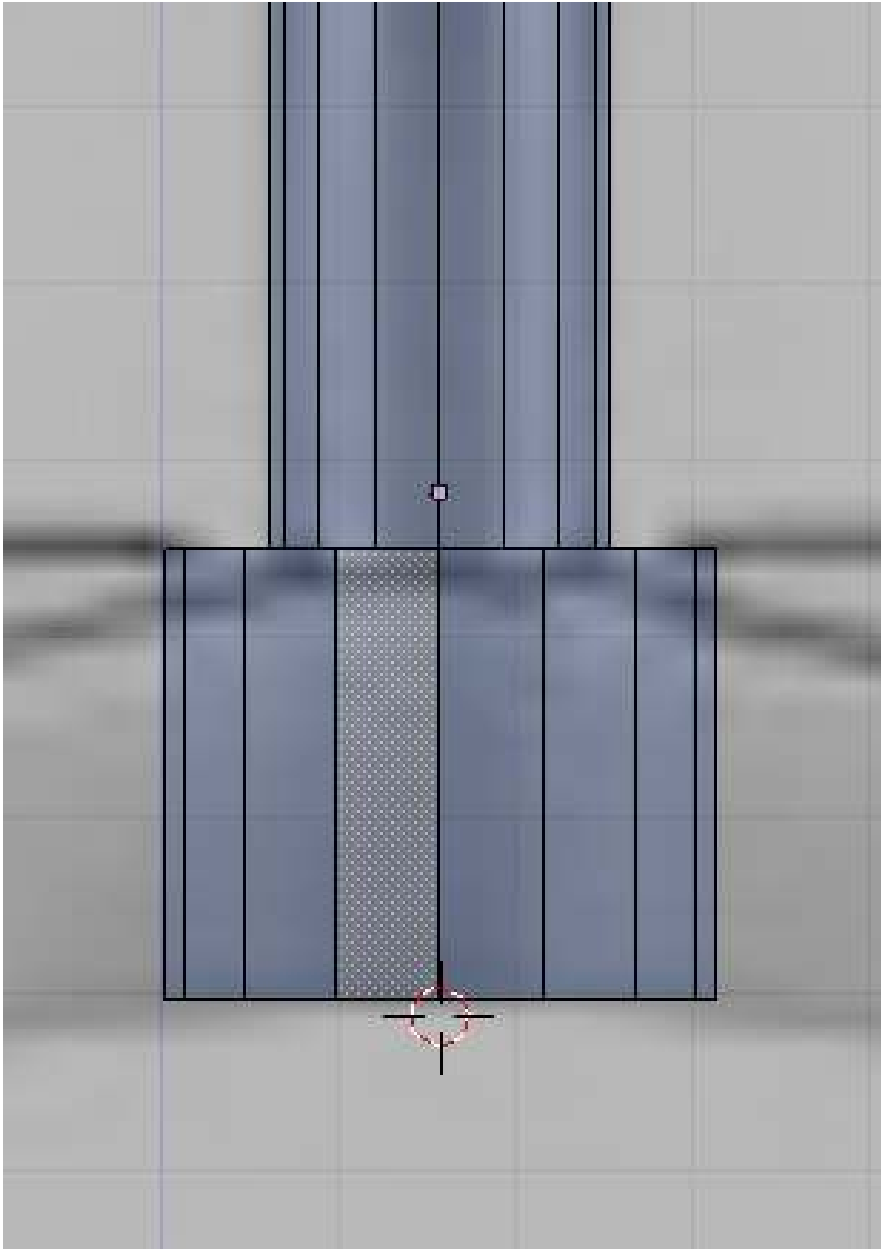


Figure 1497

In top view select the two topmost faces and extrude to form the back leg. Shape the tip according based on the reference. Here is the steps taken to shape the tip. First after extruding, Constrained scale the tip along the x-axis (SKey->XKey) to make the tip thinner. Then move the tip lower. Deselect all, then select the top vertices of the tip and lower this vertices further.

p Extruding a leg

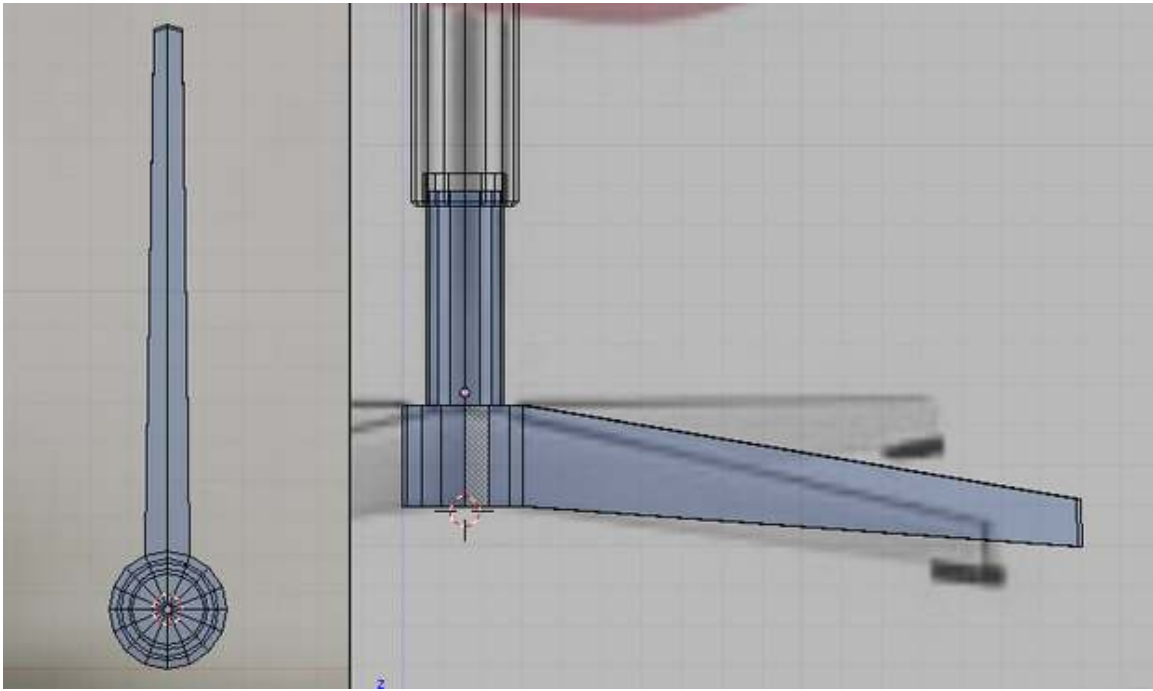


Figure 1498

Shape the base as shown bellow. This is done by selecting the vertices where the leg connects to the base then constrained scaling it along the y-axis to zero (Skey->YKey->0key). Then move the scaled vertices further back.

p Shaping the leg

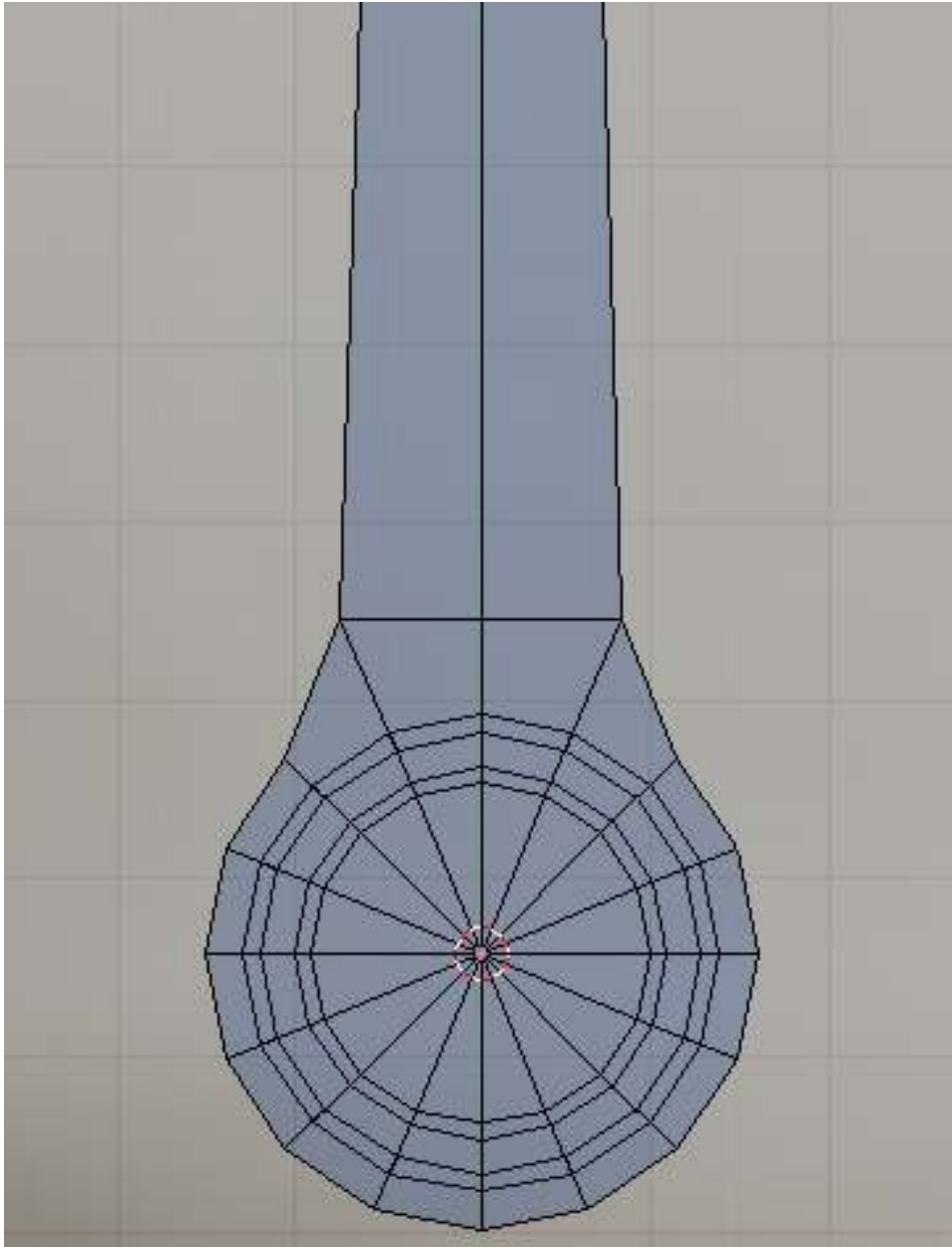


Figure 1499

Then add an edge loop as shown bellow. constrained scale the newly created edgeloop along the x-axis to make it thinner.

p Shaping the leg

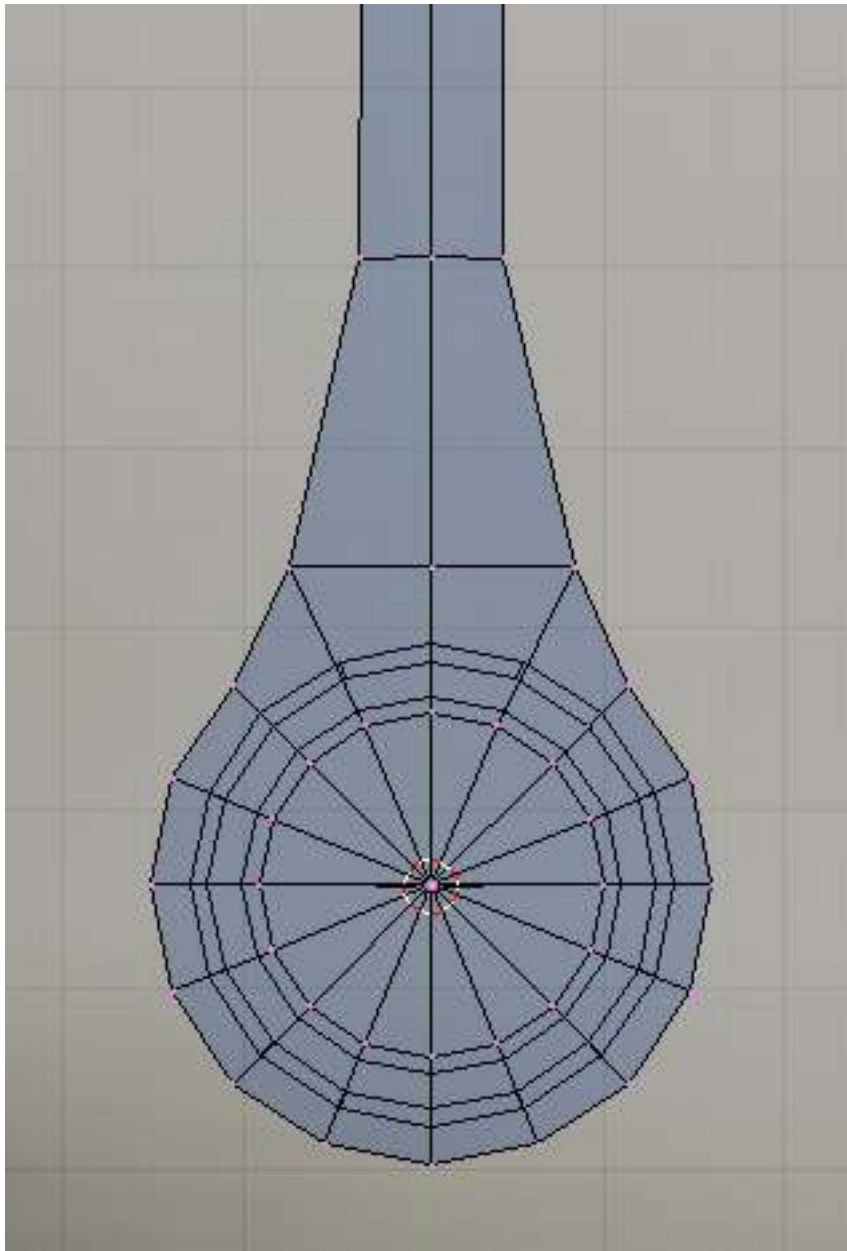


Figure 1500

We will be using the subsurf modifier to smooth our model so lets add the following edge loops so that the model will smooth properly. Add an edgeloop at the tip.

p Adding control loop

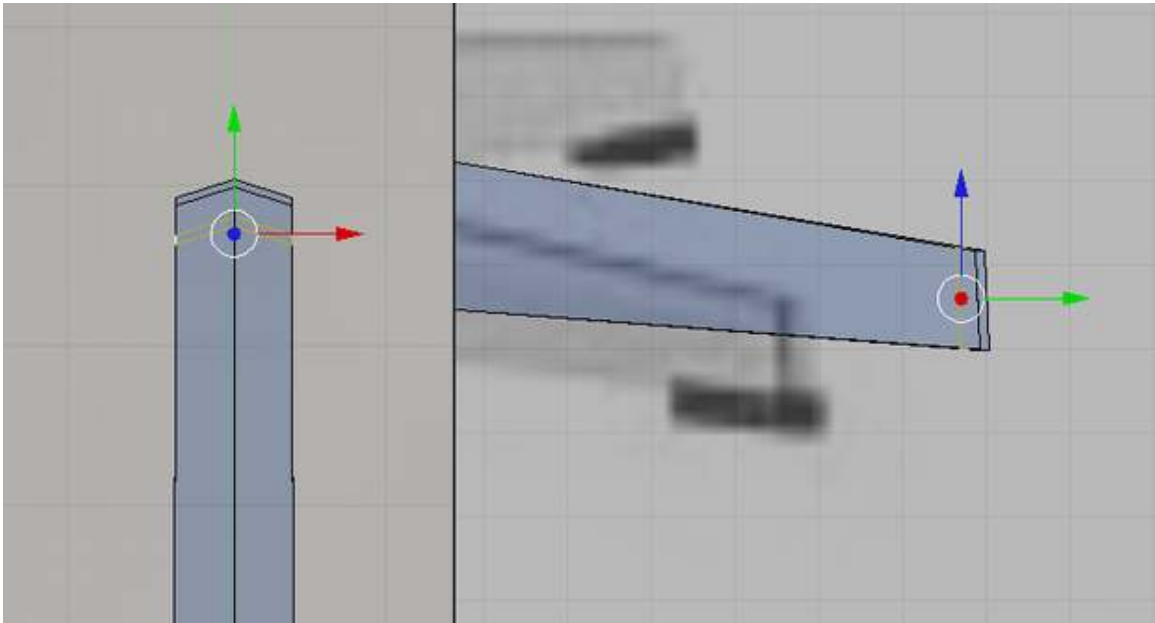


Figure 1501

And this loops on the side.

p Adding control loops

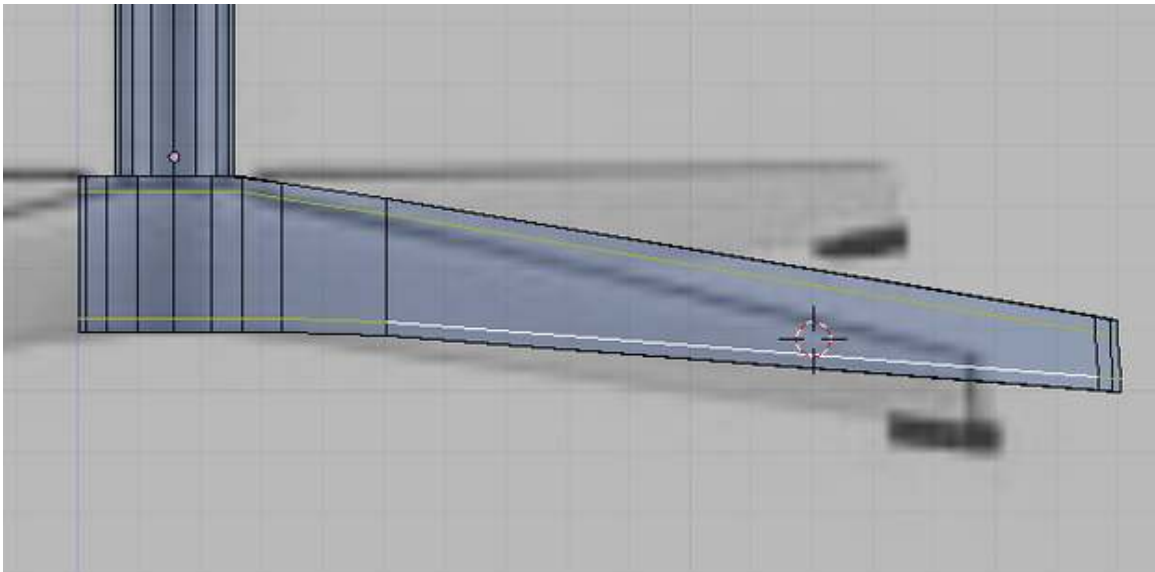


Figure 1502

Also remove the vertex that cap the top because we will not be needing it and it would cause smoothing artifact when subsurfed.

p Shaping the leg

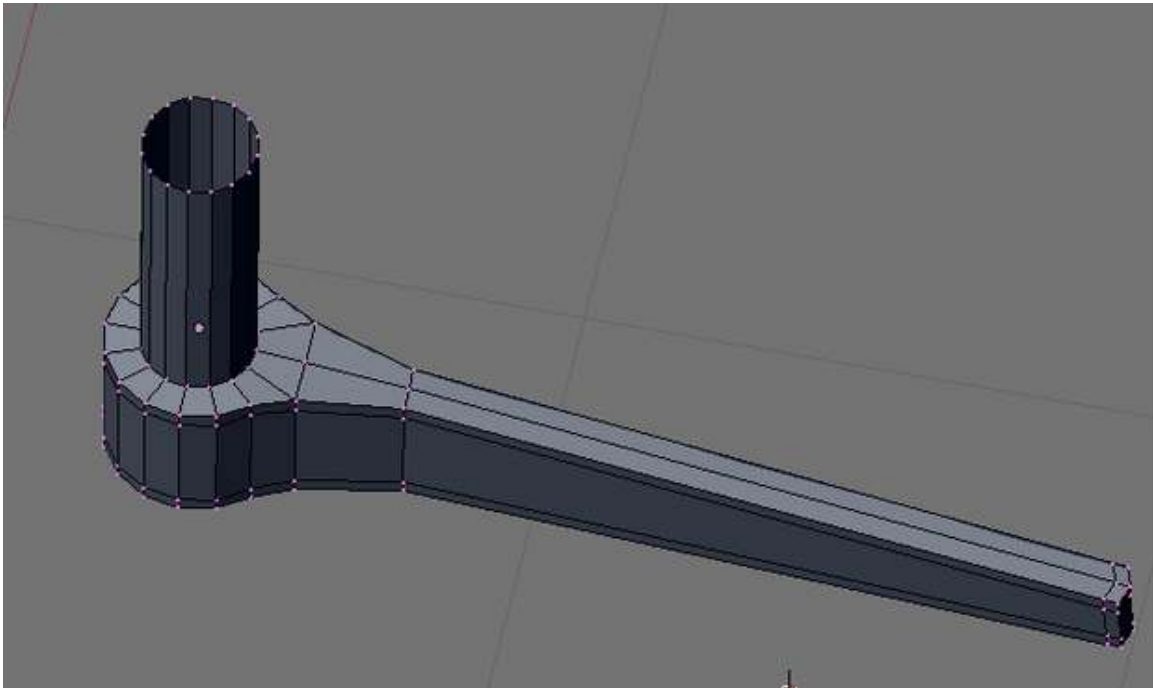


Figure 1503

Now how about the other legs? We will be using Spin duplicate tool to make those. In top view select the central vertex then snap our 3d cursor to it.

p Relocate 3d cursor to the center of the base

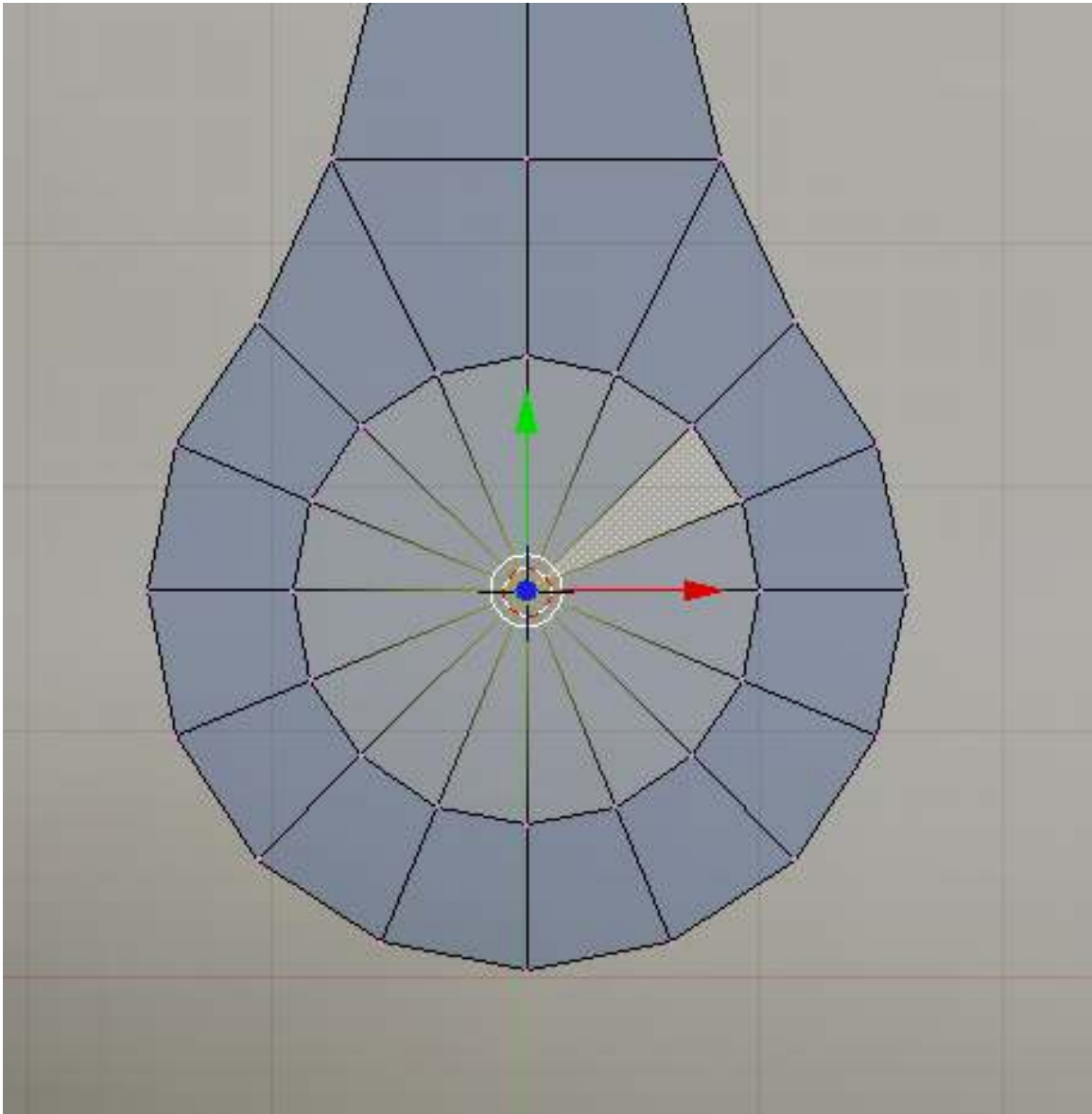


Figure 1504

Now remove the other part of the cylinder as shown. We would be left with a quarter part of the whole leg assembly.

p Delete extra vertices

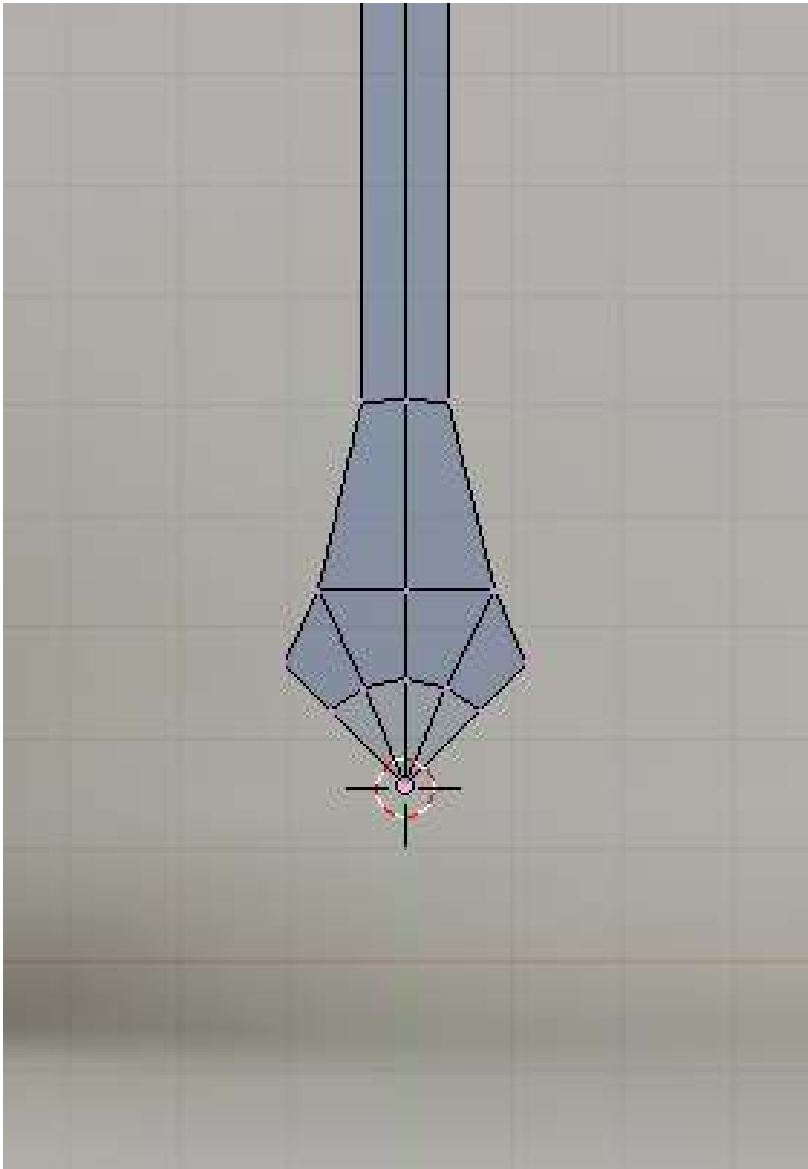


Figure 1505

Select all the vertices that are left. Change the parameters of the spin tool to this: degrees to 360, steps to 4. Click on the "Spin Dup". If you have multiple 3d views, your mouse cursor would turn into a question mark. Move your cursor to the top view and then click. The other legs would appear as shown bellow. This legs are made of separate mesh and the original leg have a duplicate at exactly the same position creating double vertices. Connect the mesh and remove the doubles by selecting the whole mesh then remove the doubles (Wkey->Remove Doubles).

p Spin duplicating other legs

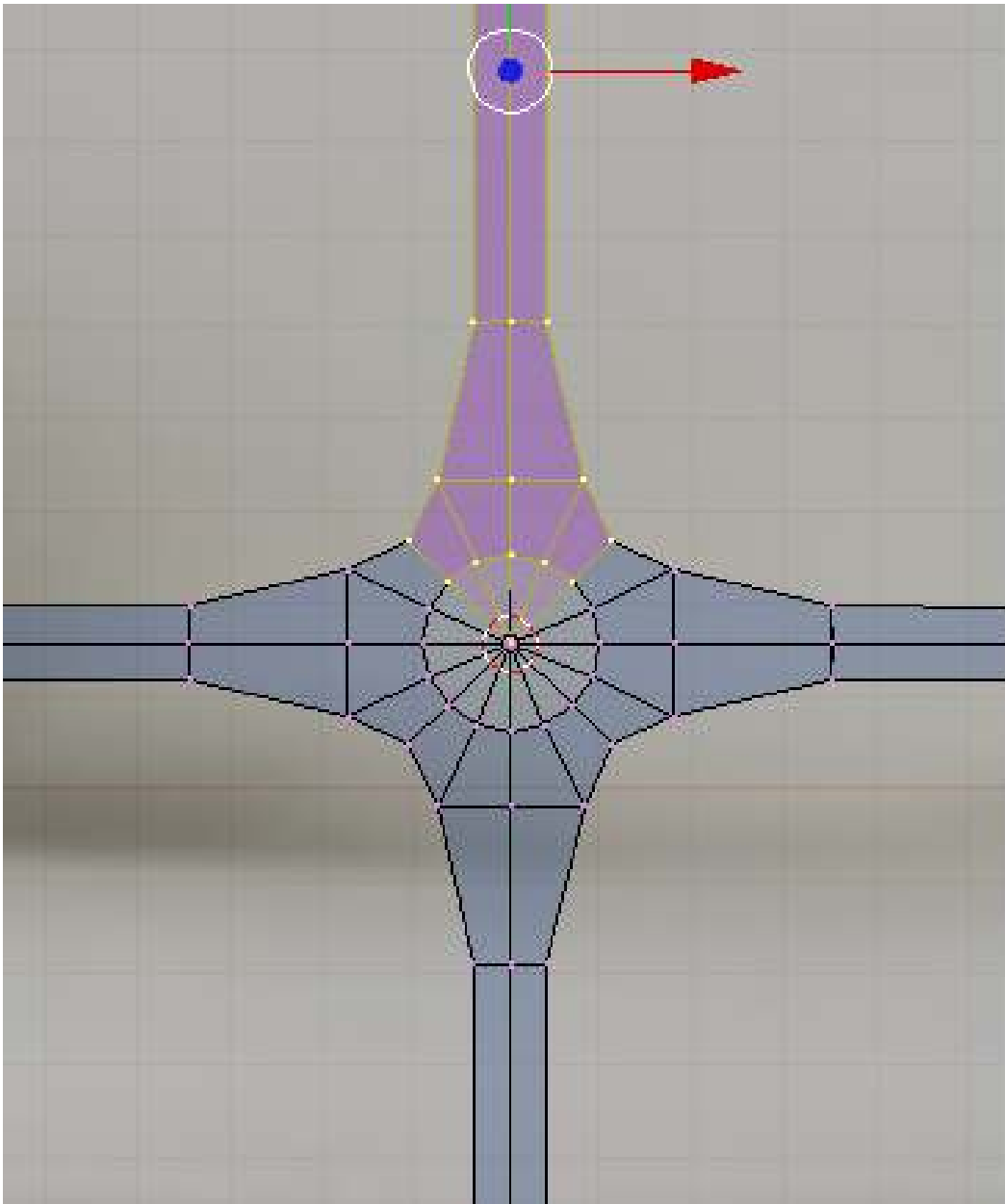


Figure 1506

233.4. The Stops

Now lets create the Black Stops on every tip of the leg. We would be creating this joined to the legs as one object then separate it later. Its just that it seems more easy that way.

Without leaving edit mode, add a cube. scale an move it to one of the tip. Move the cube so that it would be lower that usual to separate it from the other mesh making it easier to edit it without disturbing the others.

p Primitive for the stops

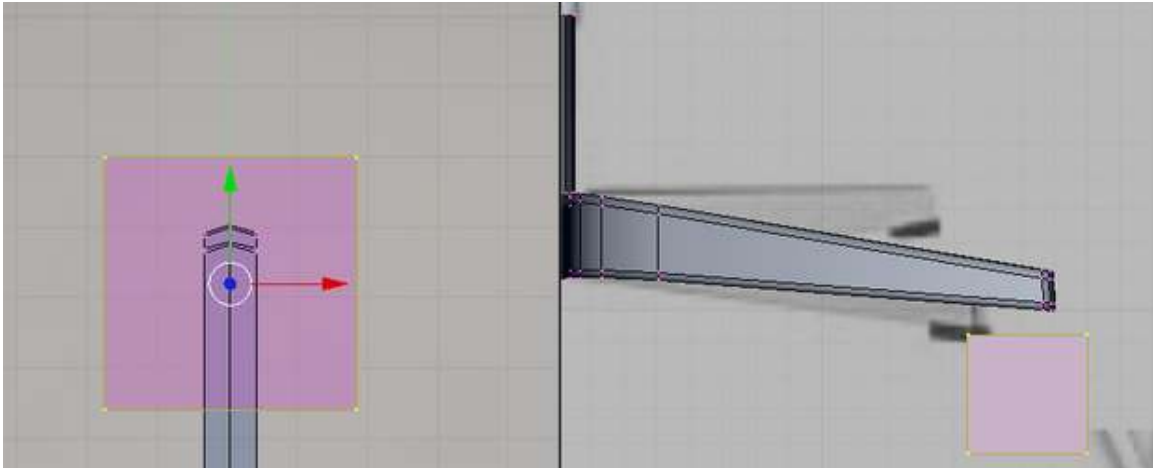


Figure 1507

Shape the cube to match the tip. This should be easy by now. The whole step is left out.

p Shape the primitive

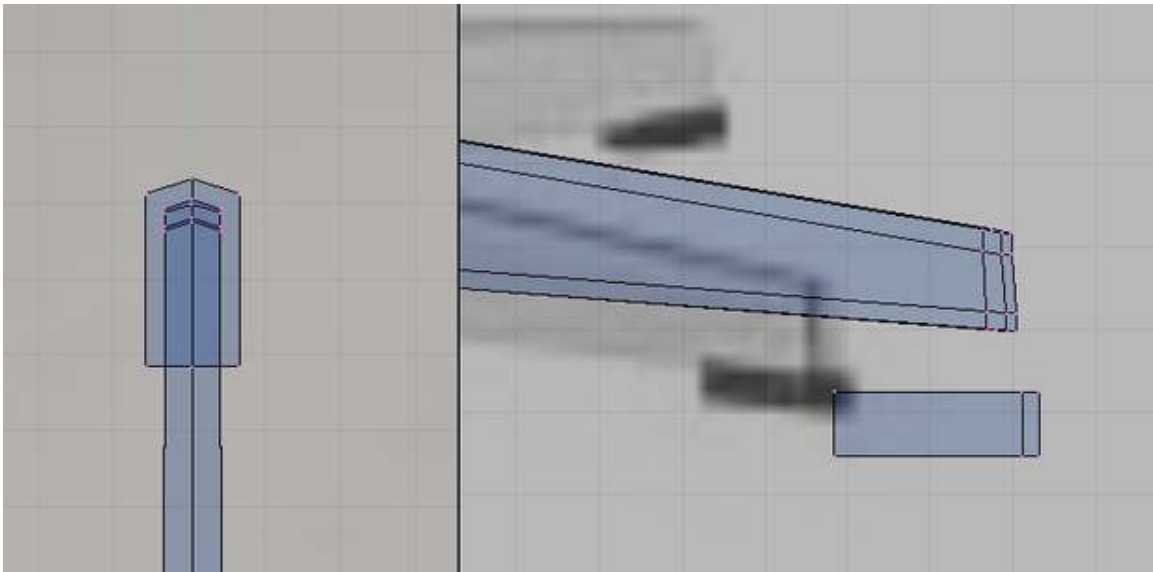


Figure 1508

Subsurf would be used to this too so add extra loops to control how the mesh would be smooth.

p Stop with control loops added

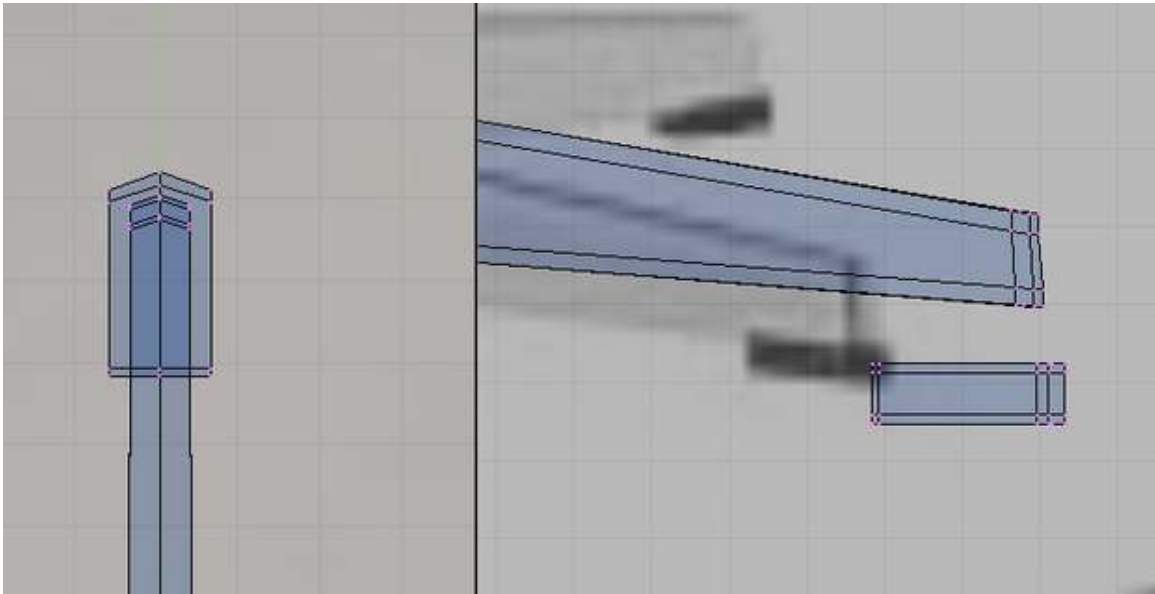


Figure 1509

Now make copies of the stops using the Spin Duplicate tool. Don't forget to remove the doubles.
p Creating the whole stops with Spin duplicate tool

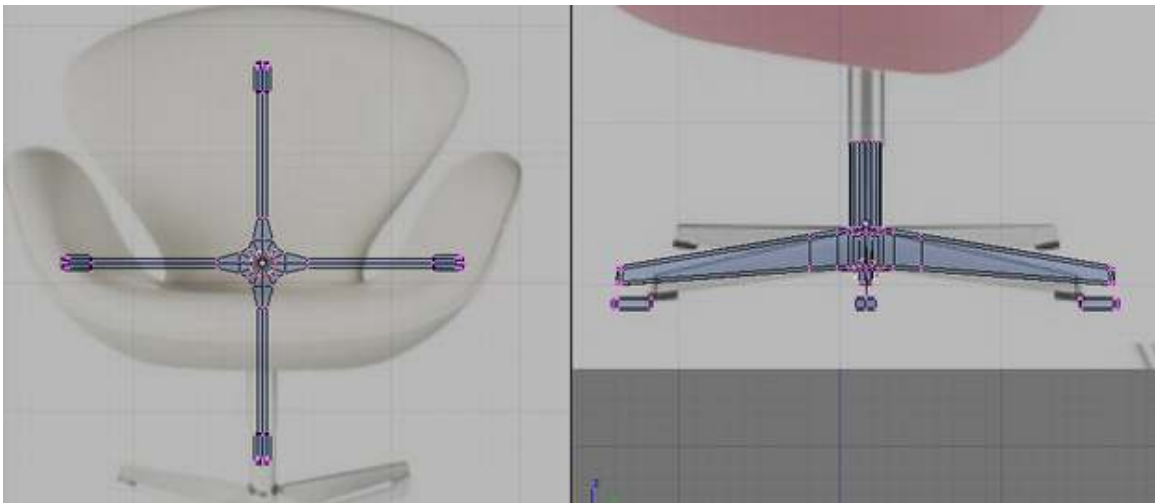
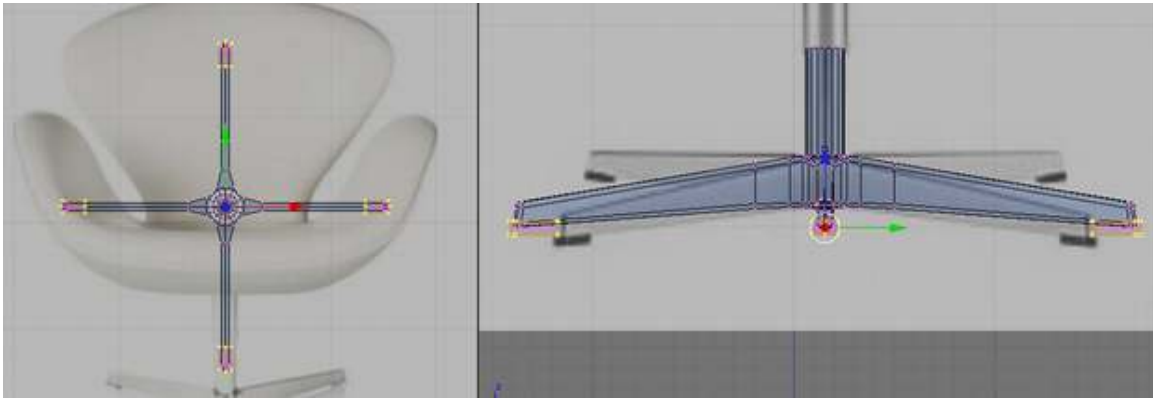
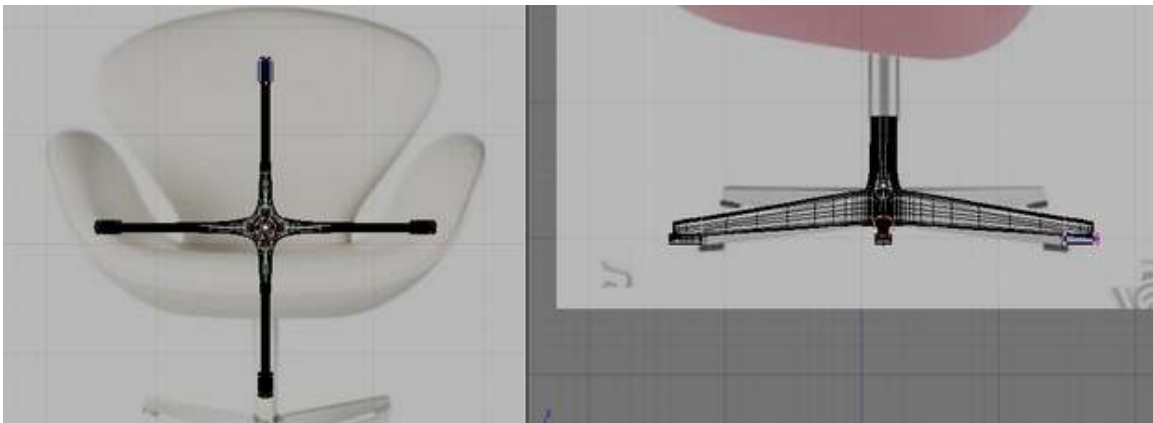


Figure 1510

Select all the tips then move them together up until they intersect with the tips of the legs.
p Positioning stops

**Figure 1511**

Now lets separate separate meshes to individual objects. Press Pkey and select "all loose part" option.
p Separating the Stops

**Figure 1512**

Notice that the other part of the mesh would suddenly change in appearance. Don't worry. It just means that those meshes are now separate objects by them selves. If you go to object view you would now be able to select the leg and the tips as separate objects.

There are some issues though when you use the subsurf modifier on the leg base as shown.

p Unwanted smoothing effect

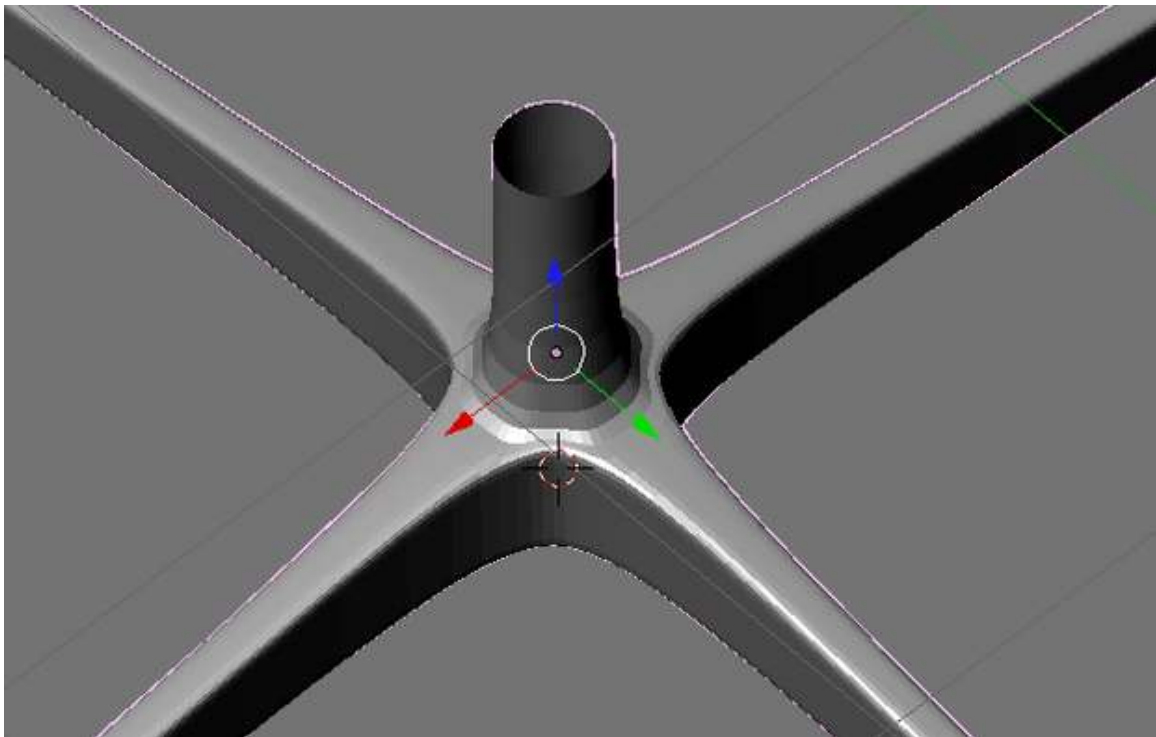


Figure 1513

Select the leg and go to edit mode. Add the following edge loop.

p Adding control loop

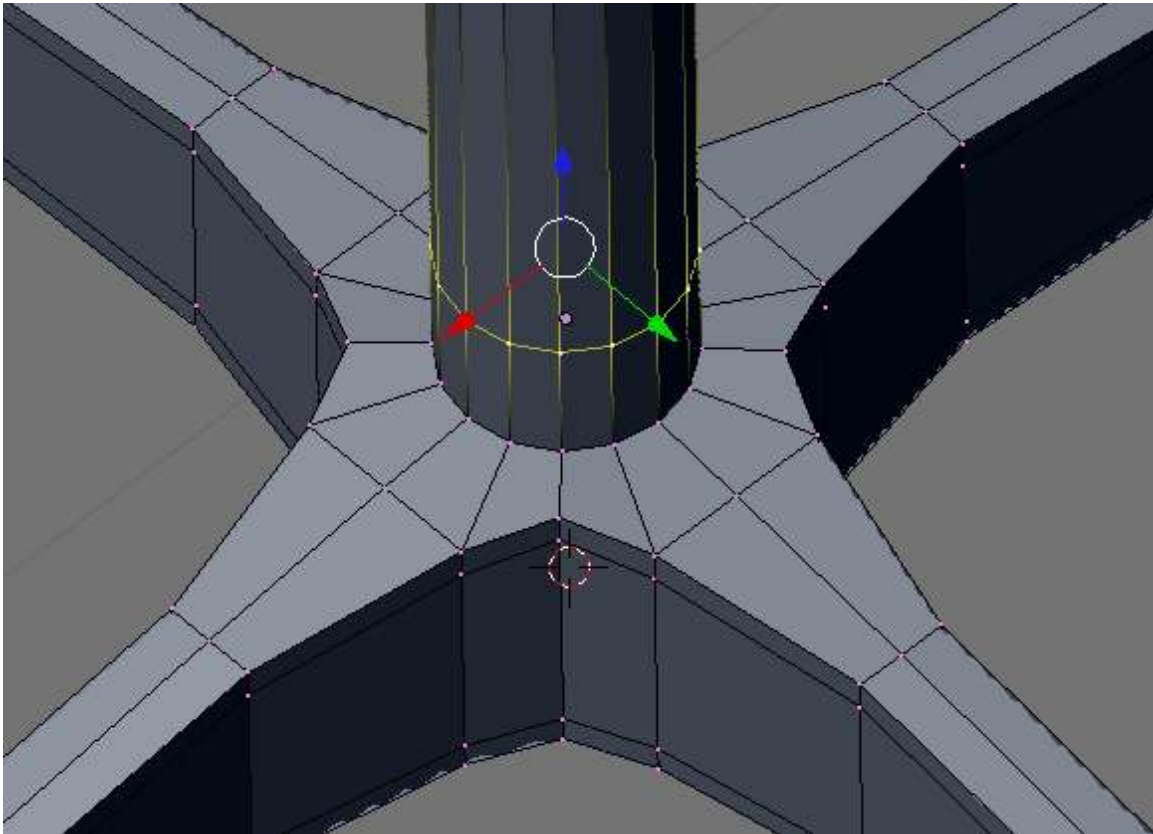


Figure 1514

Scale the base loop too as shown.

p Scale lower loop

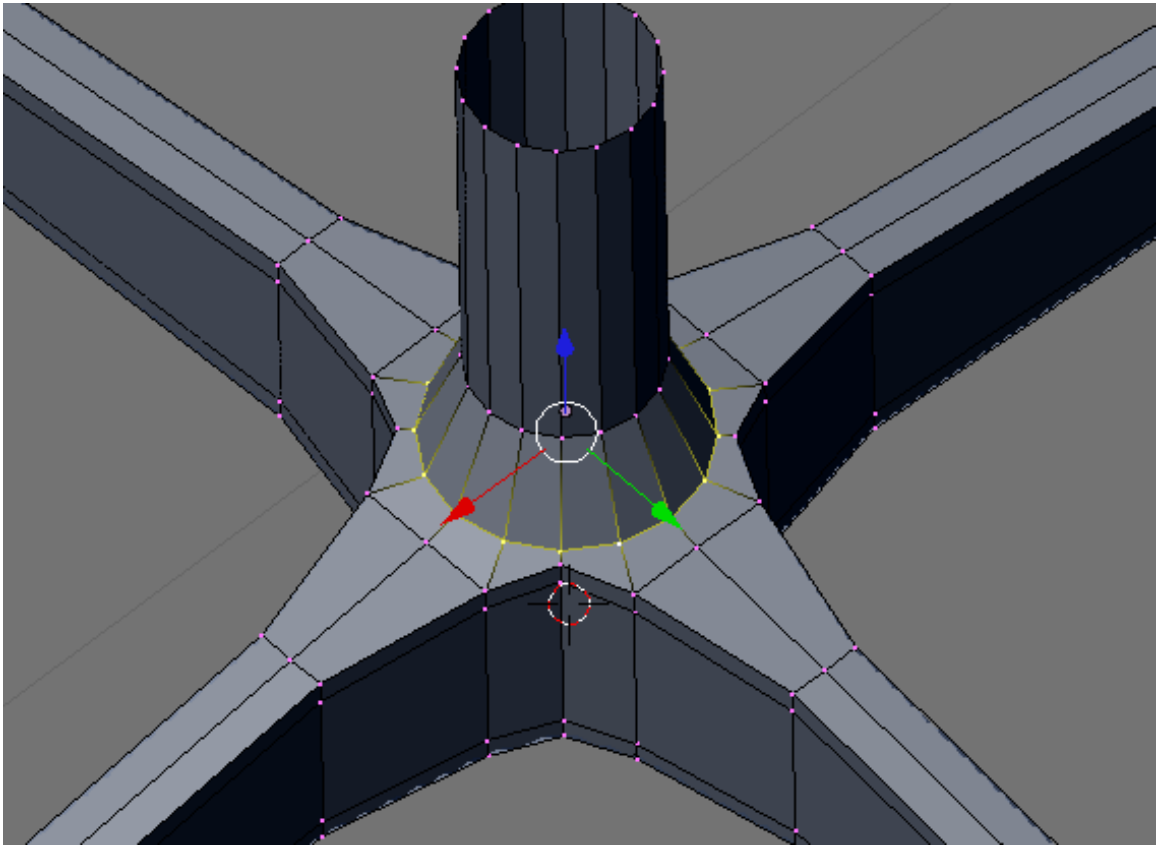


Figure 1515

The issue should now be solved.

p Much better

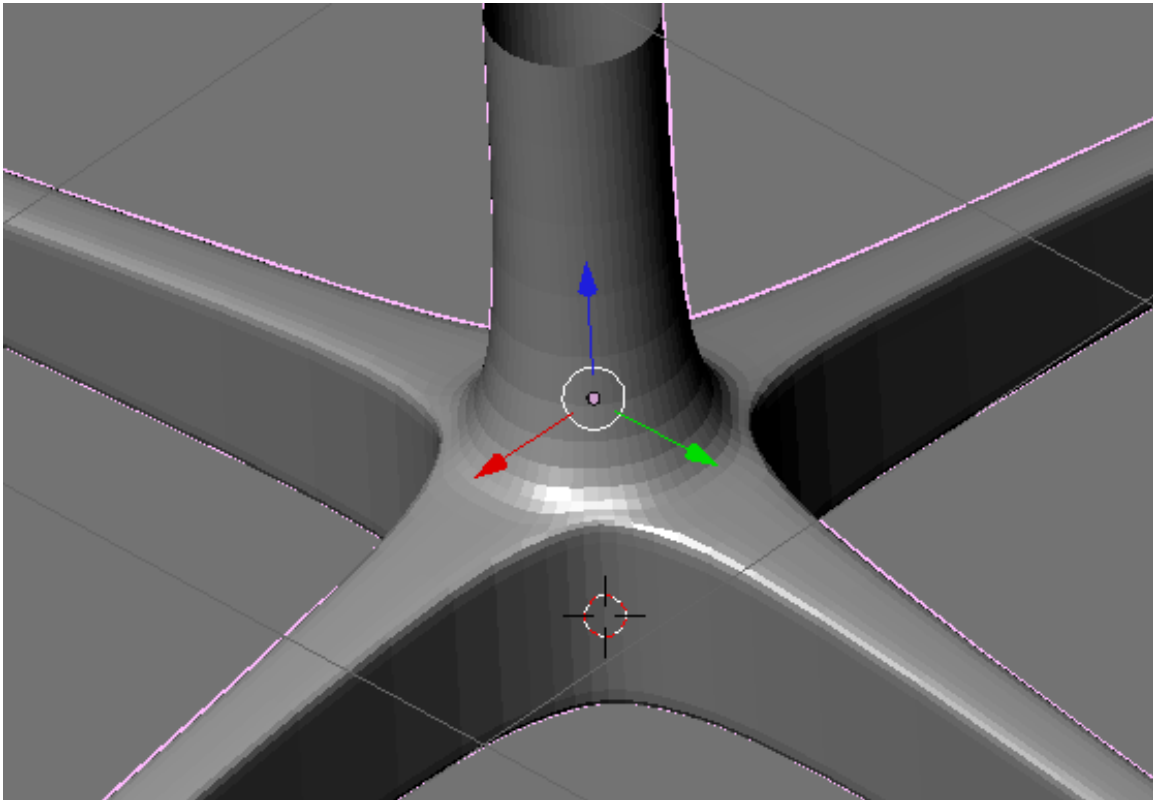


Figure 1516

This is how the whole leg assembly should look. All have subsurf modifier on.

p Leg assembly



Figure 1517

There is still an issue though. This is not important in most part but will be taken care of for this time.

When one of the leg stops is selected in object mode, it would show that its center is not at the center of the mesh. The center is where the gizmo (the red, blue and green arrow) would be located.

p object center not at the center of the mesh

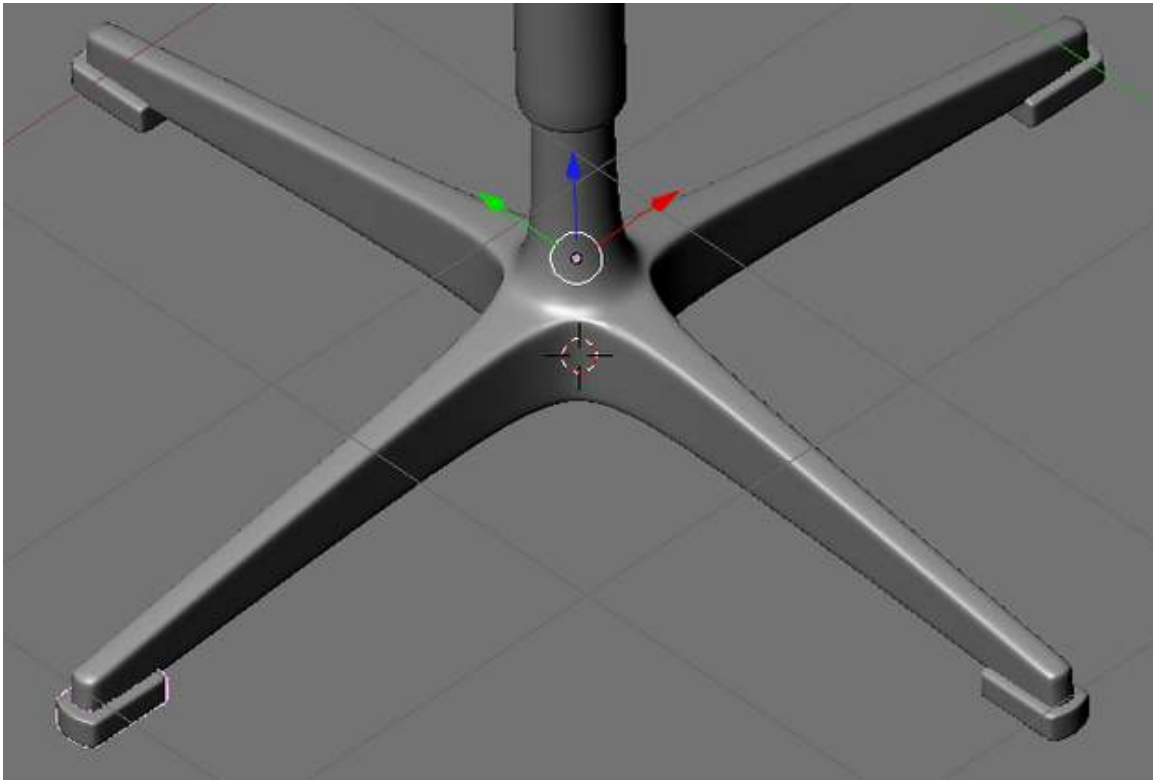


Figure 1518

To correct this select the offending object. Click the Mesh menu (located at the header) and navigate to "Transform" and select "Center New" from the options. Blender would automatically calculate the new center based on the objects mesh. Now when the object is selected, the center would be located at a much more appropriate location.

p Center at proper place

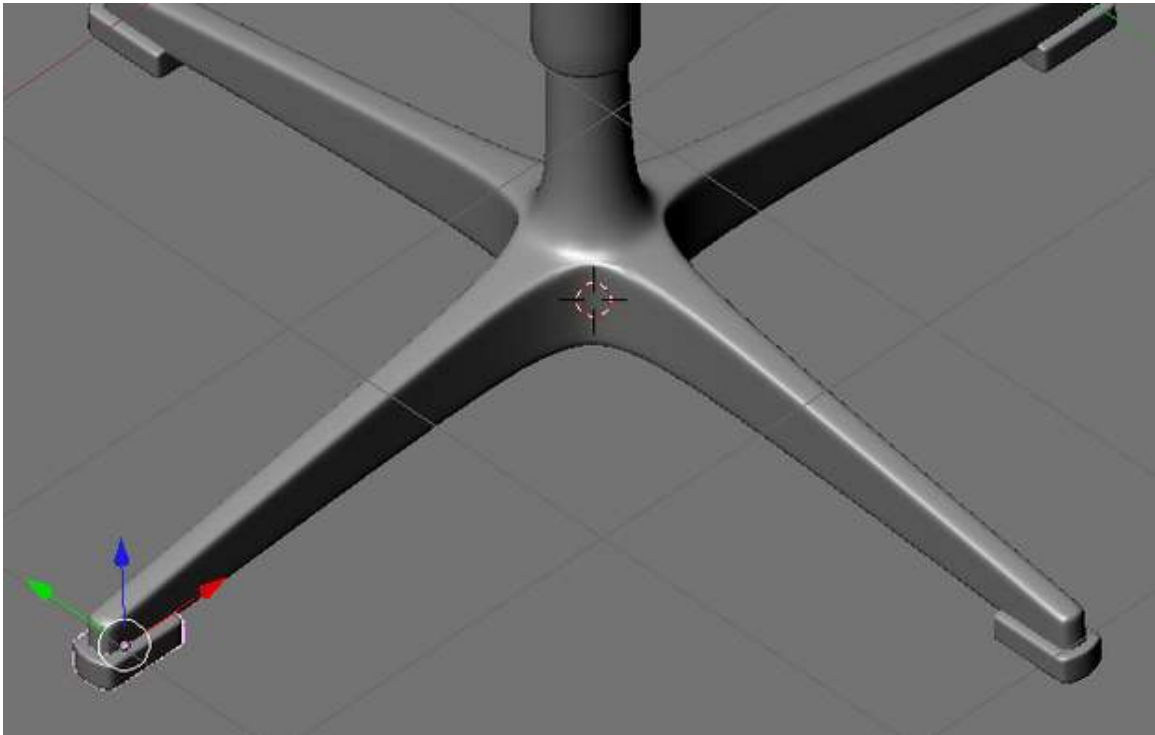


Figure 1519

Do the same to the other stoppers.

Now if you haven't named each individual part yet, You can do it now. Name each part sensible names like uppertelescope, lowertelescope, etc.

233.5. The Whole Chair

Unhide or Show the layer where your chair seat is so that it would unite with the legs. Now this is how the whole model should look like.

p Finished Model



Figure 1520

234. Illustrative Example: Modeling a Simple Human Character

p Simple human figure



Figure 1521

We will be modeling this very simple human figure. This time many details are left out to save time and space. It's assumed that enough knowledge is gained from previous tutorials for readers to understand this stage. Shortcut keys are also left out. Read first the chair modeling tutorial if this comes out too difficult to read.

235. Modeling a Human Character - Preparations

Actually, this model is made without any background reference images but enough knowledge of human figure makes this possible. Modeling the human body or any model in that case requires knowledge of it beforehand. Even with the background image references, there are still some details (like the armpit) that is not easily visible or are difficult to comprehend with just simple orthogonal projections. It is advised to make some study on basic human figure and proportions first.

For those who likes to have a background reference image to use while modeling, use this orthogonal views of the model. You could still try and test your knowledge on the human figure later by trying to model one without image references.

236. Modeling a Human Character - Modeling

237. Blocking the Figure

As with many models done in box modeling, lets start with a cube. The model is symmetrical so lets make use of it. Add a loop running vertically at the middle and delete the vertexes on the left leaving you with only half of the cube. Now add a mirror modifier.

p Initial box

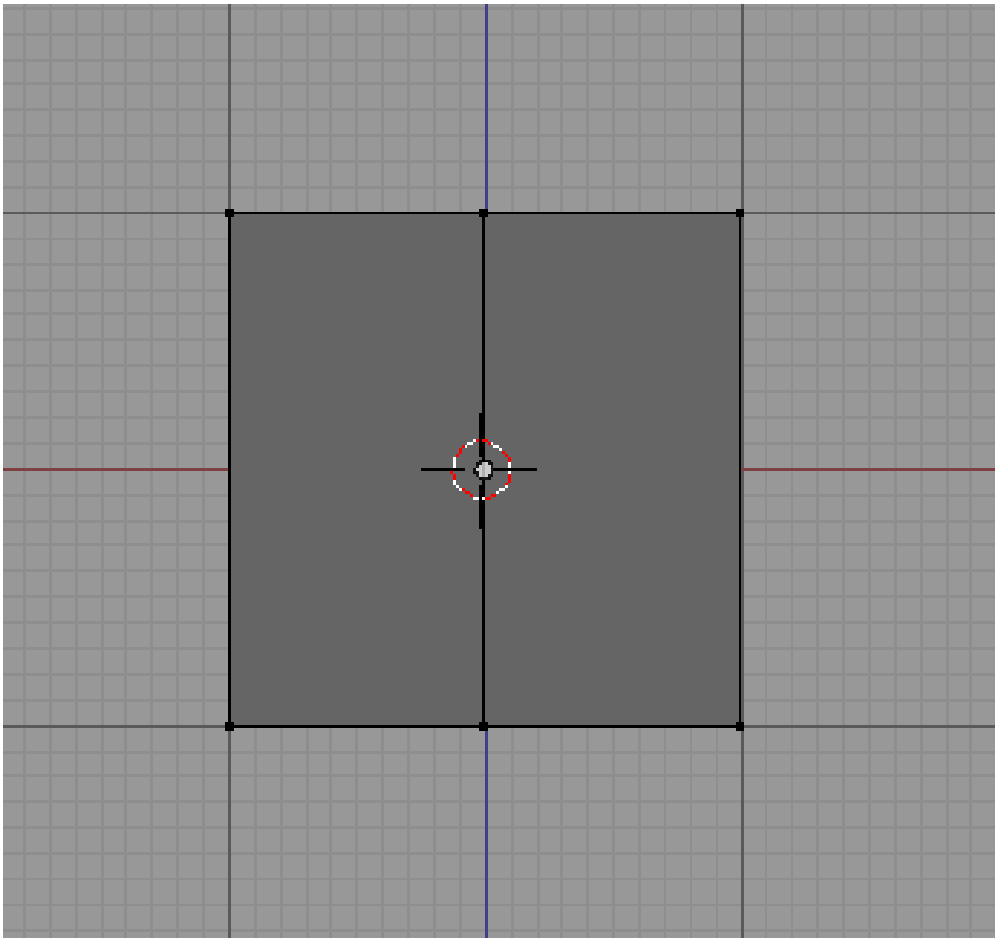


Figure 1522

Now add a loop running horizontally and fashion out the torso.

p Forming the torso

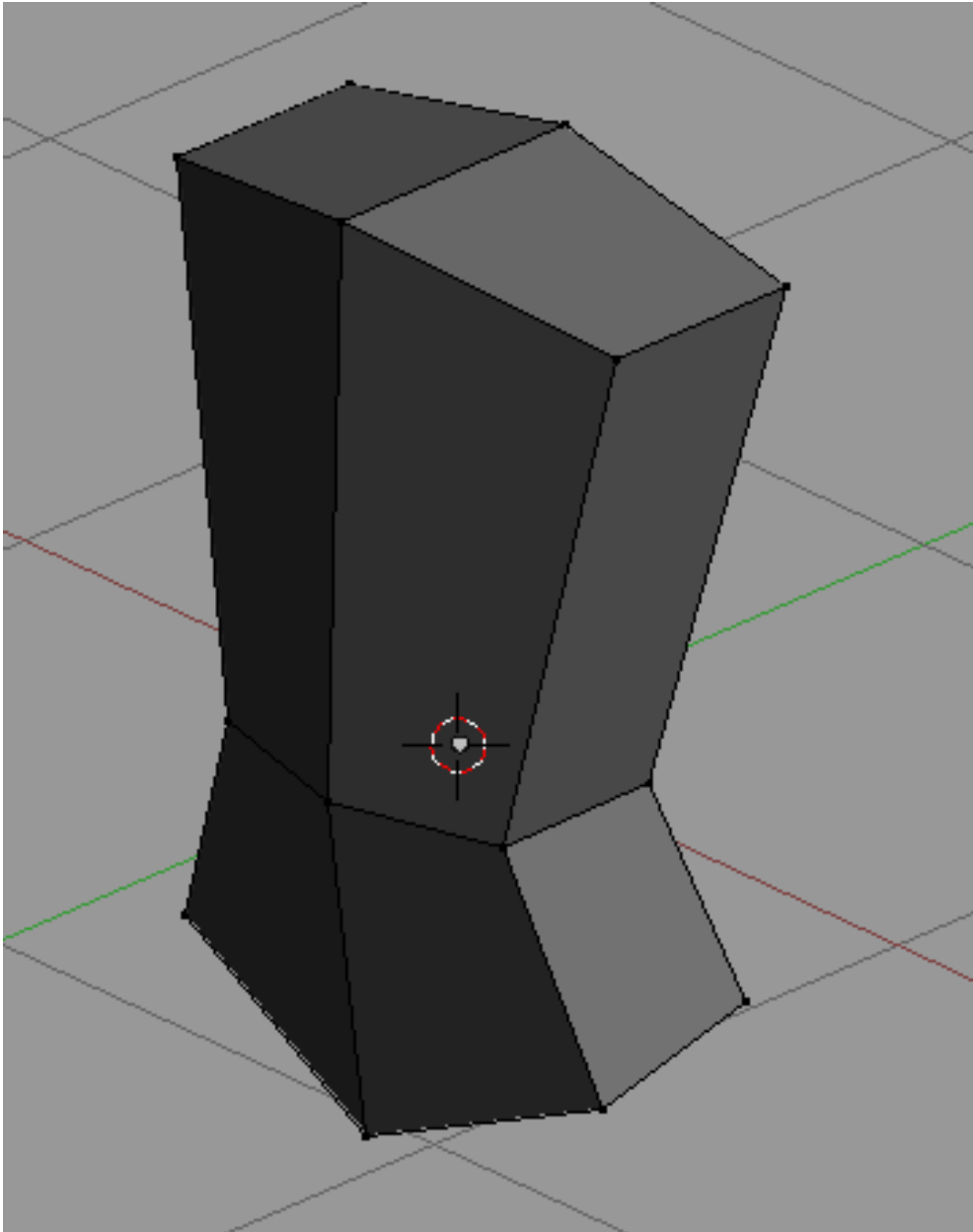


Figure 1523

Add a vertical loop on the torso so that we will have more geometry to work on.

p Adding more geometry

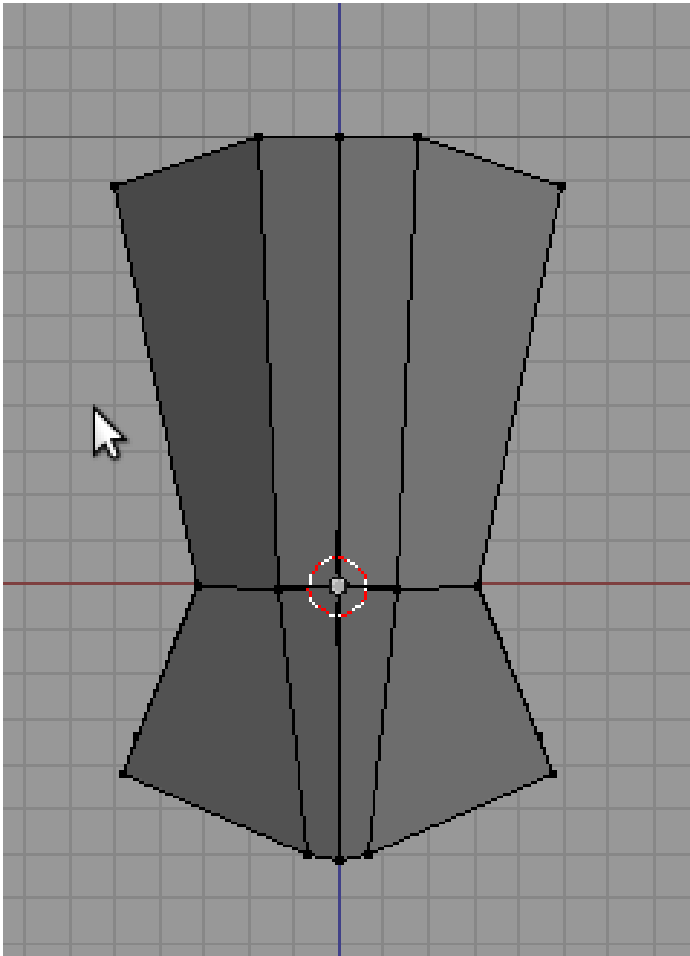


Figure 1524

Extrude the head and the legs. Add loops around the head and the bottom of the legs and extrude out the head and feet.

p Extrude head and legs

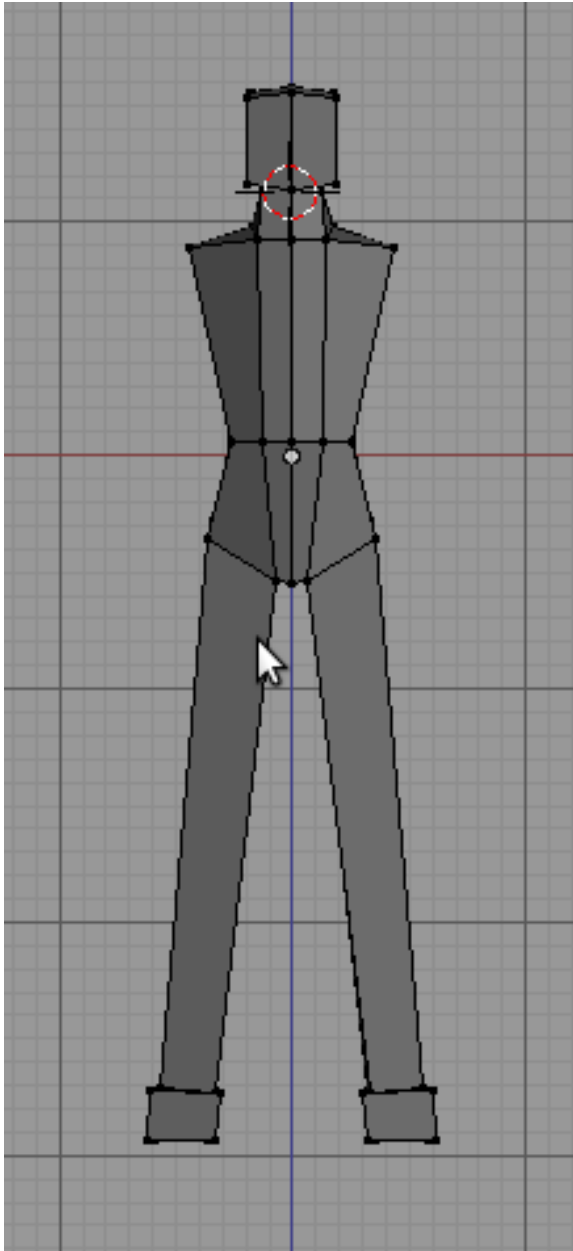


Figure 1525

p The model at another view

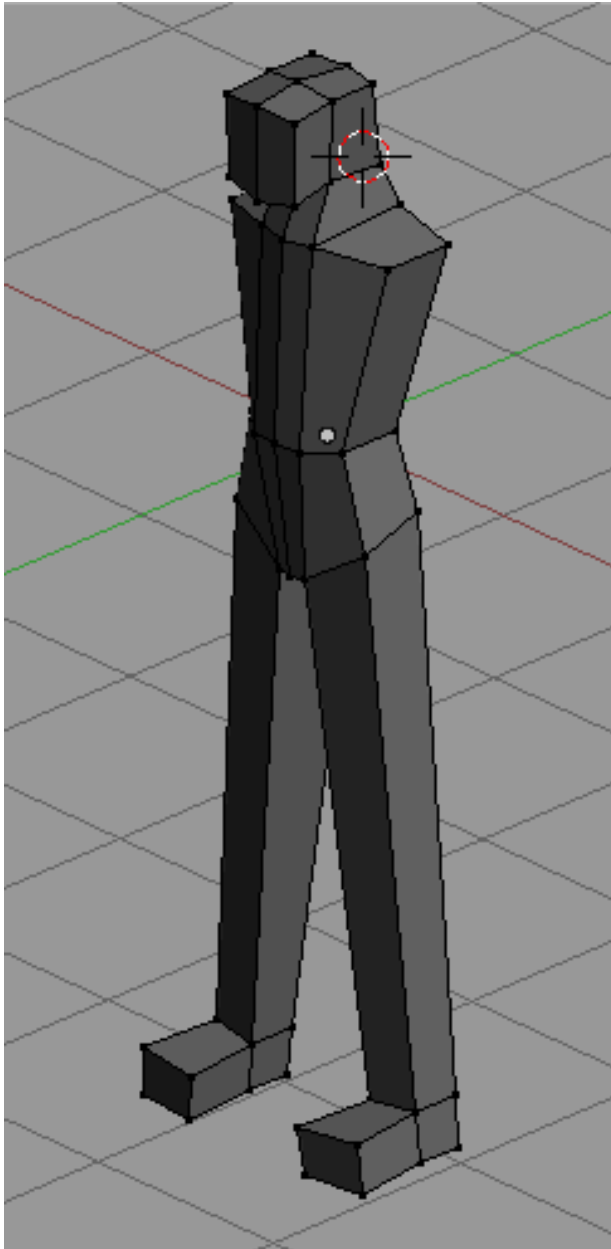


Figure 1526

Now add a loop around the torso and extrude out the arms. This is all for blocking
p Extruding out the arms

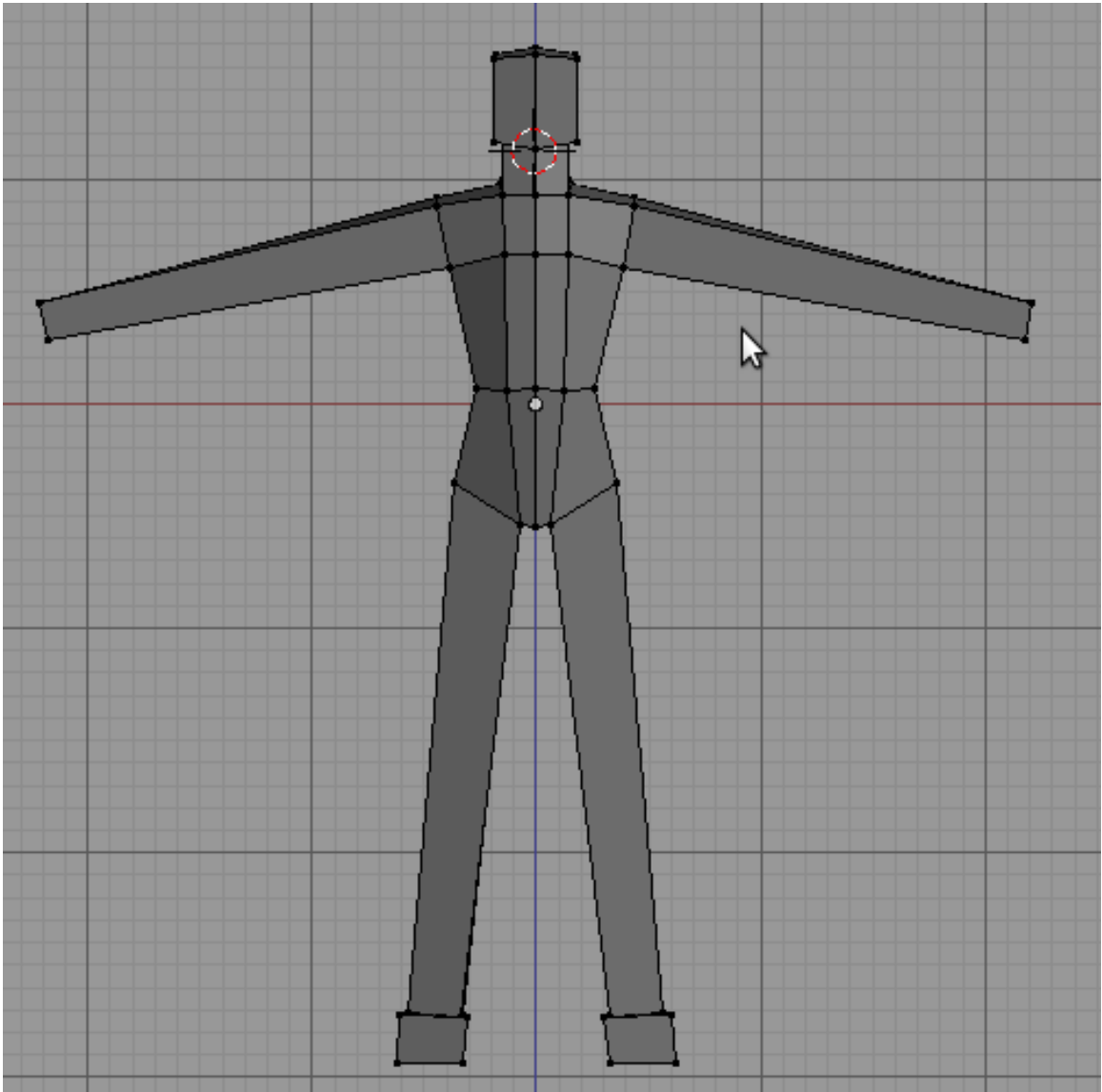


Figure 1527

p From another view

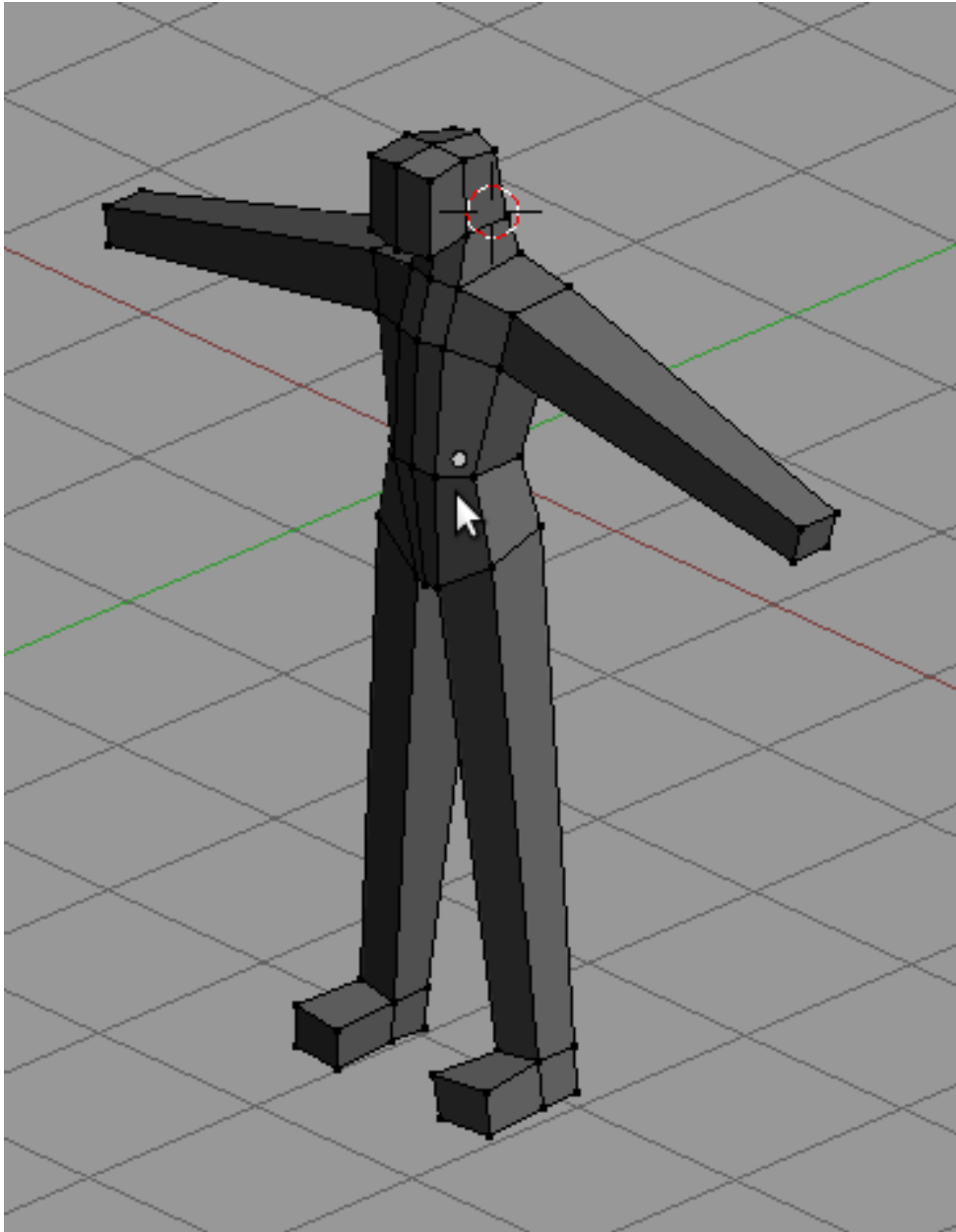


Figure 1528

238. Detailing and Finishing

Add a loops that goes through the torso and the arms and another at the sides. This geometries will enable us to round and shape the figure.

p Adding more geometry and rounding the figure

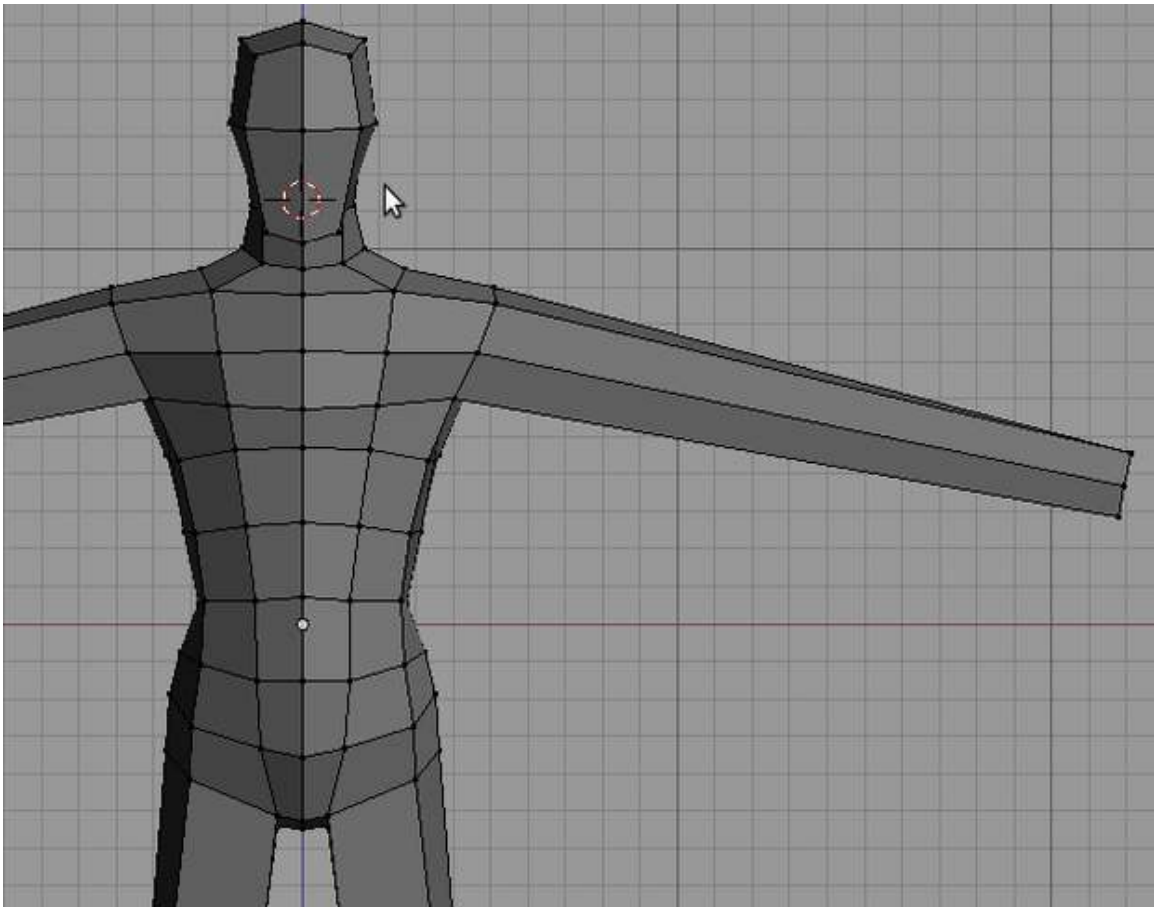


Figure 1529

Extrude out to form a rough hand. Add an extra loop around the hand and extrude out the thumb.

p Extruding the hand and thumb

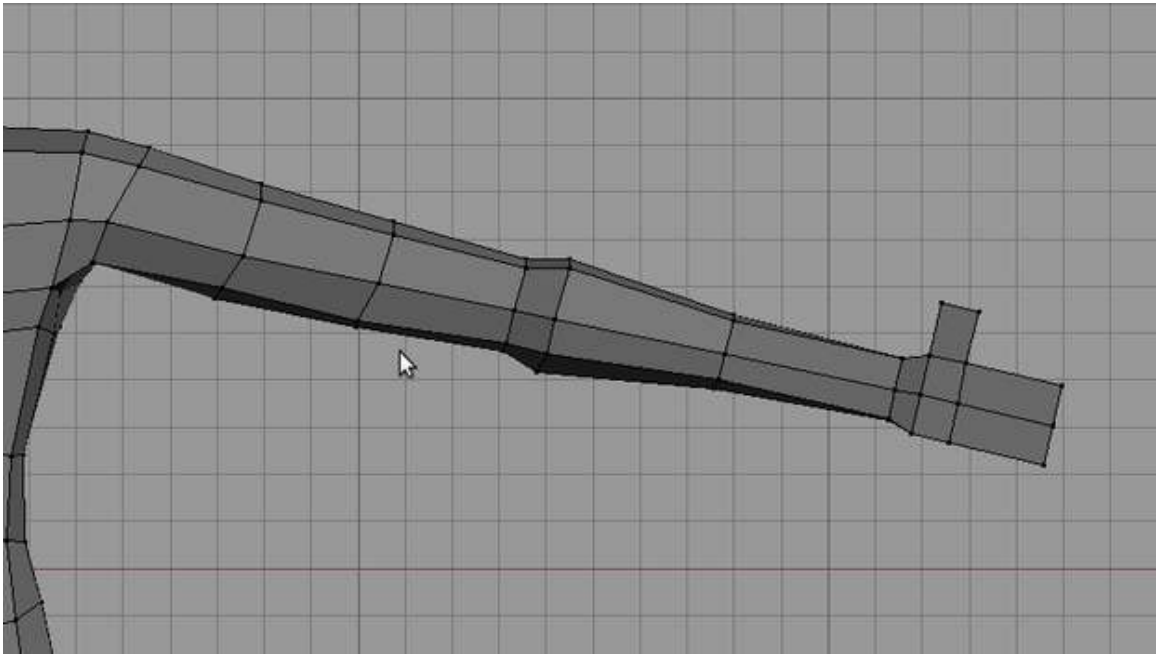


Figure 1530

Add more loops around the arm, thumb and hand. Work out this additional geometry to shape the parts.

p Shaping the hand

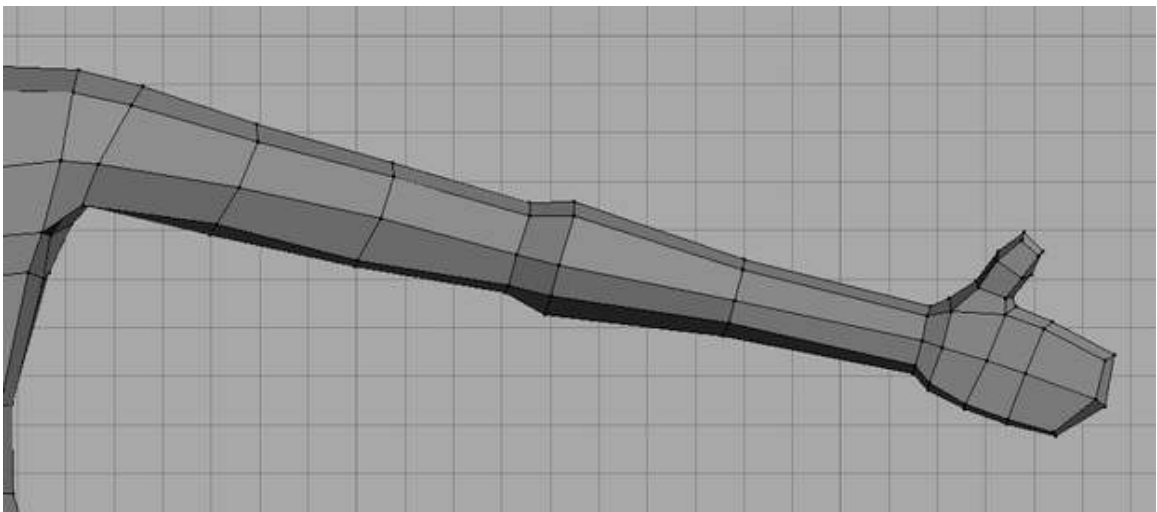


Figure 1531

Add this loop to add more geometry on the legs and to the torso.

p Adding geometry

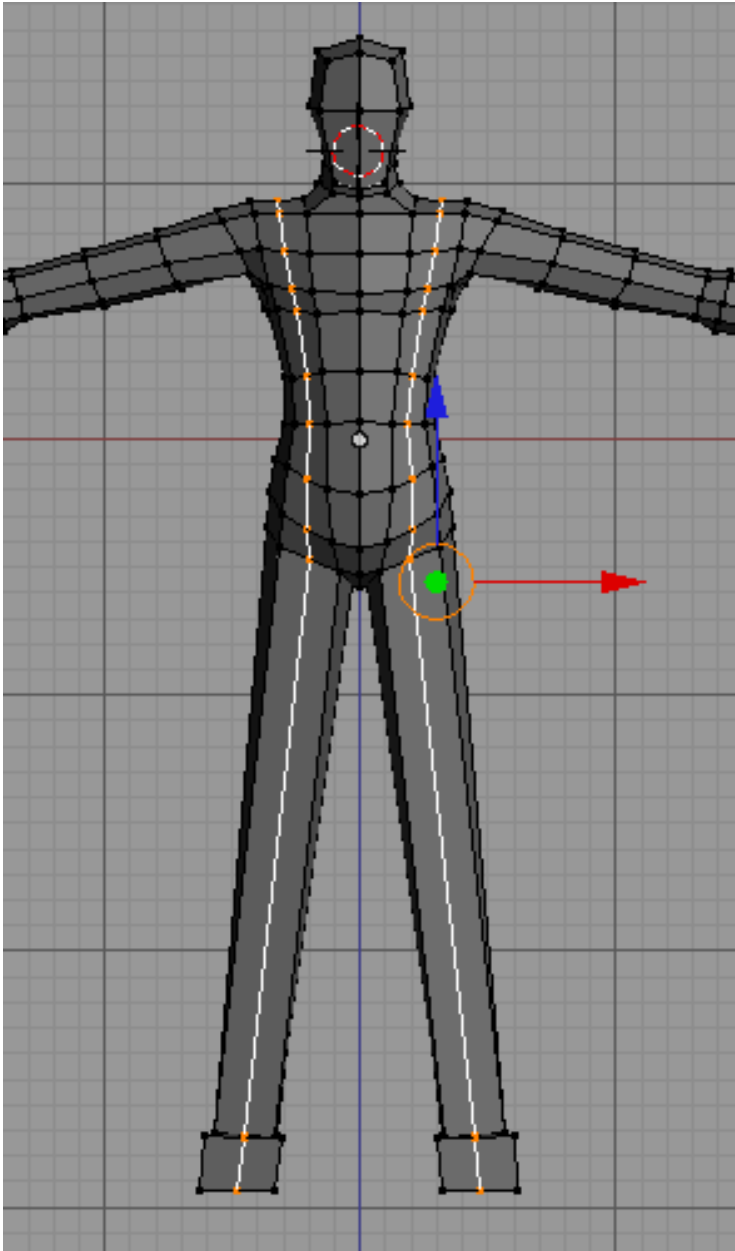


Figure 1532

And then shape.

p Shaping

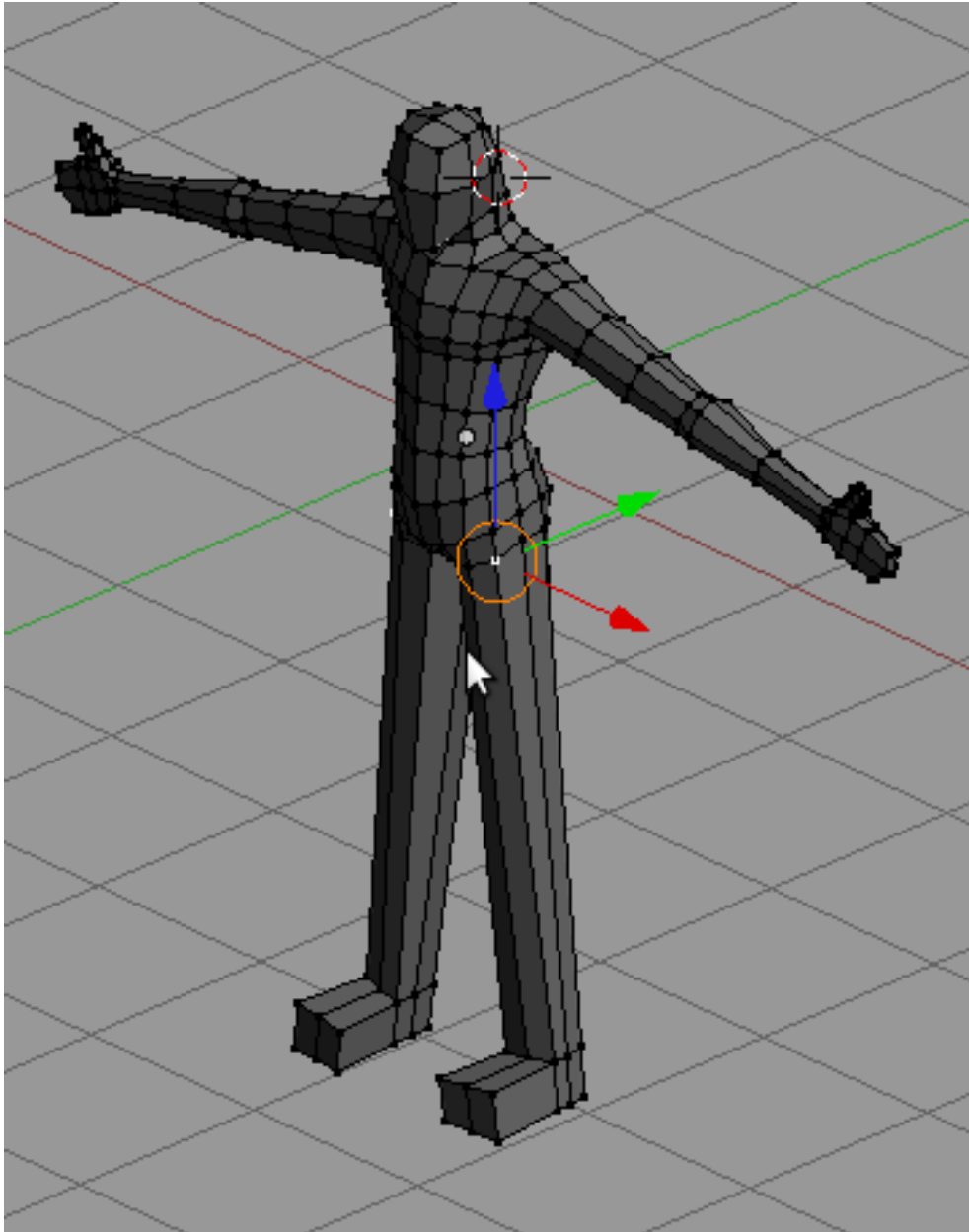


Figure 1533

Add more loops on the leg and shape out the form.

p Forming the legs

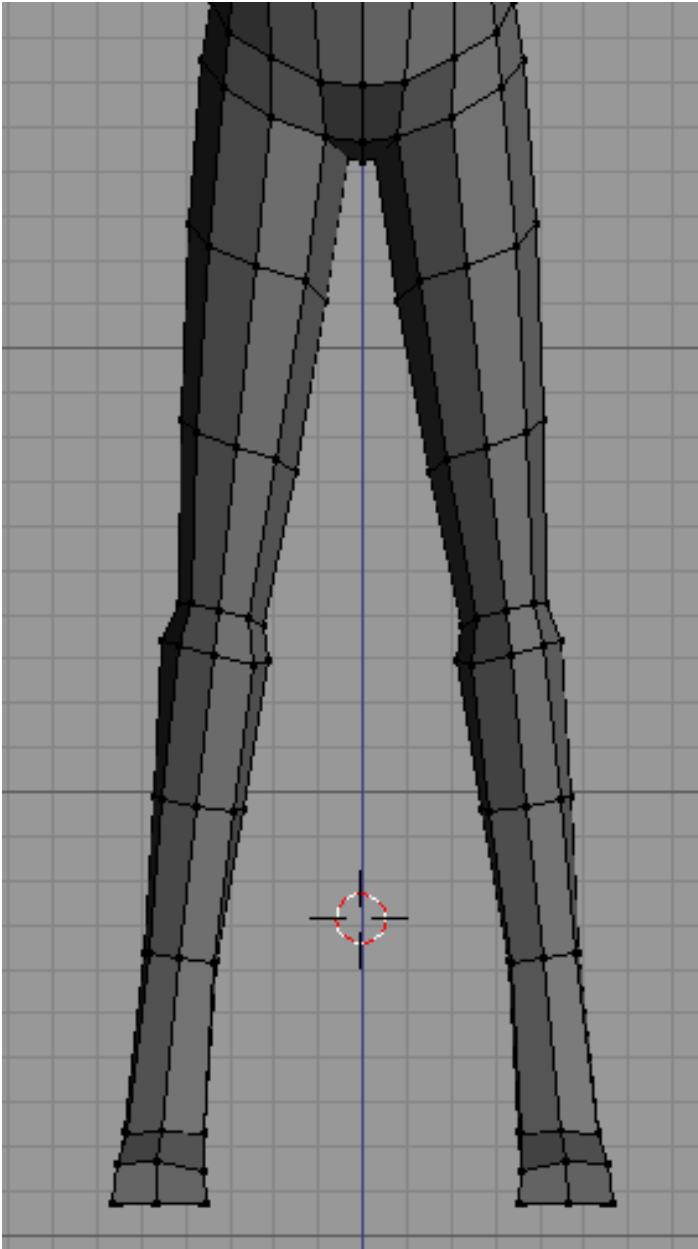


Figure 1534

Now time to shape the foot add more geometry and shape out the foot
p Initial form of the foot

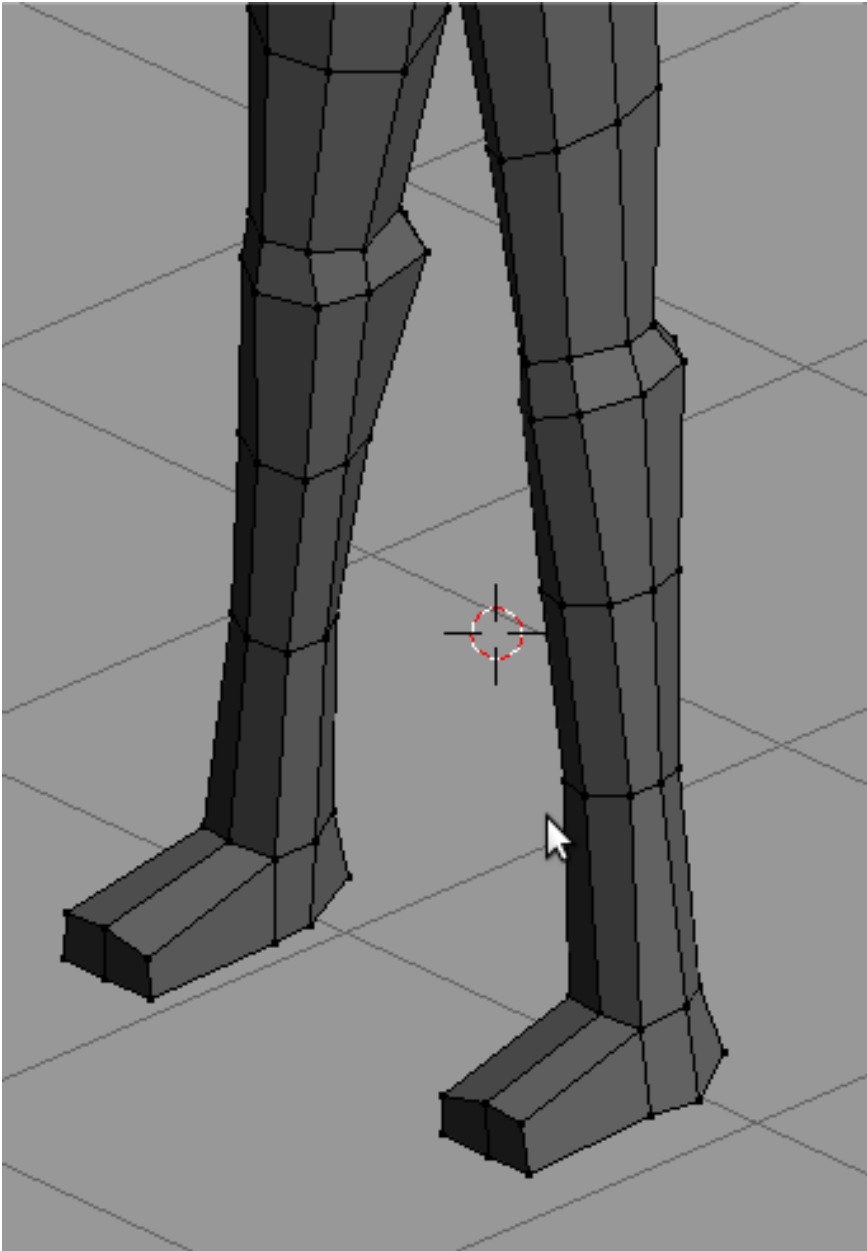


Figure 1535

p Final form of the foot

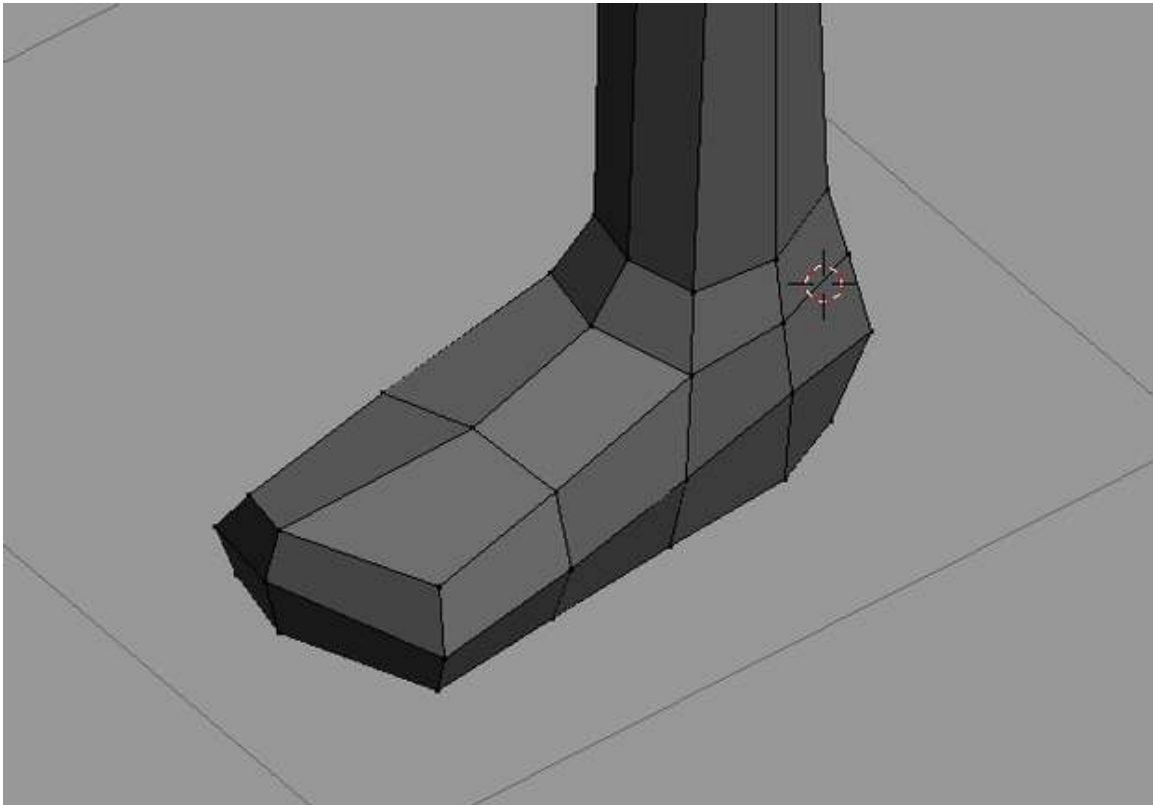


Figure 1536

Now take a look at your model. Adjust the form until satisfied. Work finished.

p Doing finishing touches



Figure 1537

239. Polygon by Polygon modeling

Although quite demanding, many modelers savor the control poly-by-poly modeling allows. However, its demanding nature makes many modelers, beginners and experts alike, to avoid it. Still, poly-by-poly modeling, if properly used, is a powerful tool.

239.1. Before we start

For this work flow, being ready is of great importance. Have several references of your subject, especially a front and side view. Also, have minutes of study on the subject and try to make out an approach in modeling it. Being ready is crucial for this.

p Image used as a reference available at the commons



Figure 1538

p Loop plans for the model

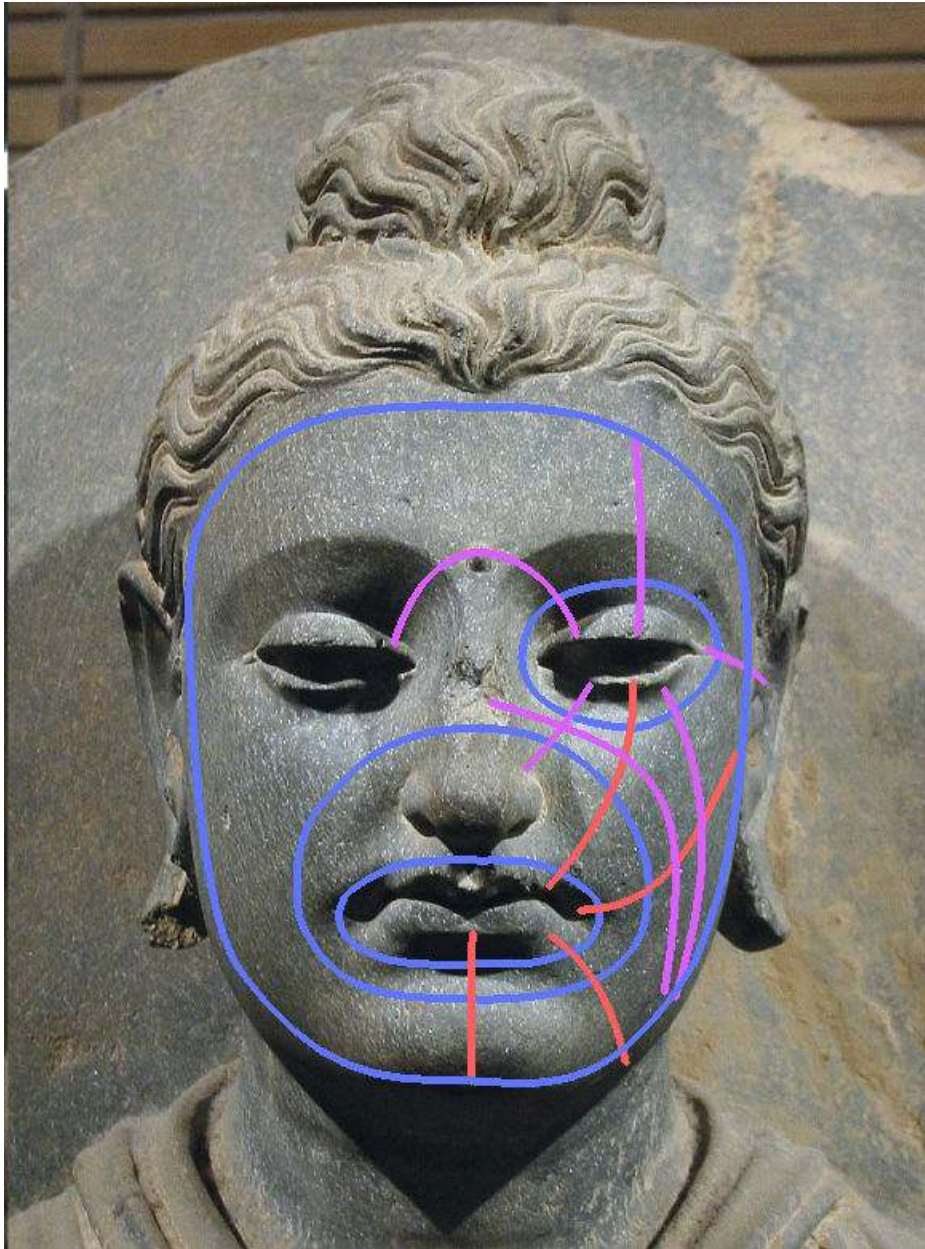


Figure 1539

239.2. The Workflow

Poly-by-poly modeling provides the most free modeling workflow ever. This freedom, however, causes confusion on how to approach modeling the subject. Searching the internet for tutorials on modeling the face using this method, you will find hundreds and each will be different from each other. Some start with the eyes, others with the mouth, others the ridge of the nose. And then each will proceed differently from that. No wonder its confusing.

Even though poly-by-poly modeling is such a free form method and constitutes a very vogue way to start and continue, a general guideline is provided in the hope that it will help modelers in using this approach.

239.2.1. To Start

Poly-by-poly modeling rather sets you free on choosing how and where to start. The only requirement is that you have a geometry, a vertex or a plane to start with. Many beginners are stumped by this. Somehow such liberty leaves them undecided.

p Starting geometry

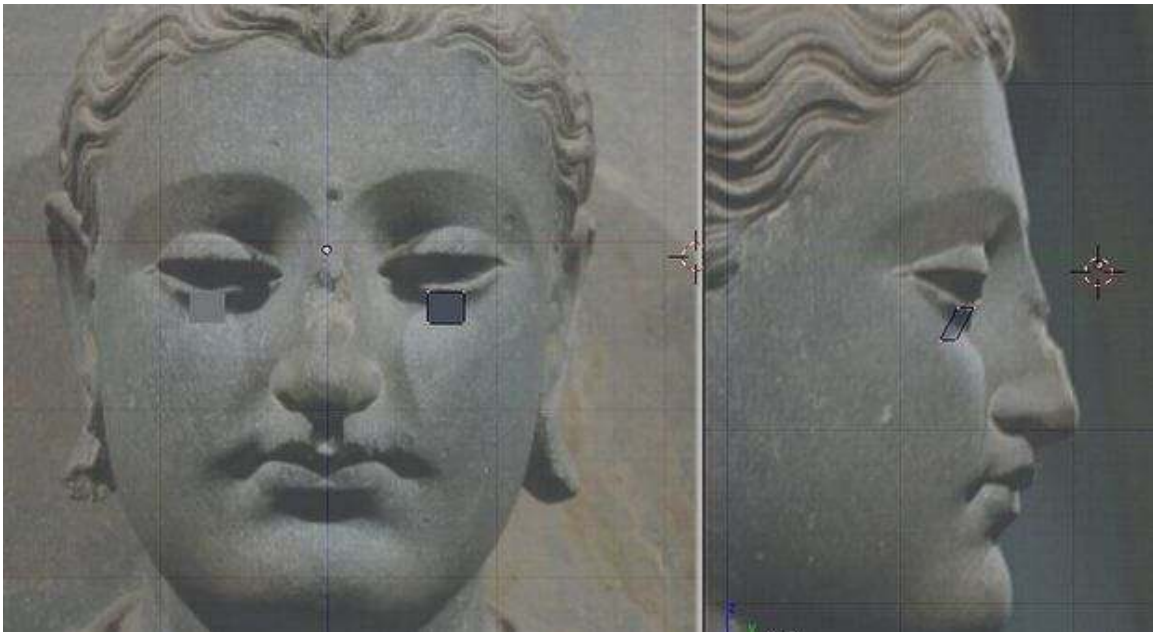


Figure 1540

A very good approach is to start by creating the important loops first. For example, in modeling the face, you could start by creating the eye, mouth and face loops. With the loops in place, it is easy to build other geometries around them.

p Starting with important loops

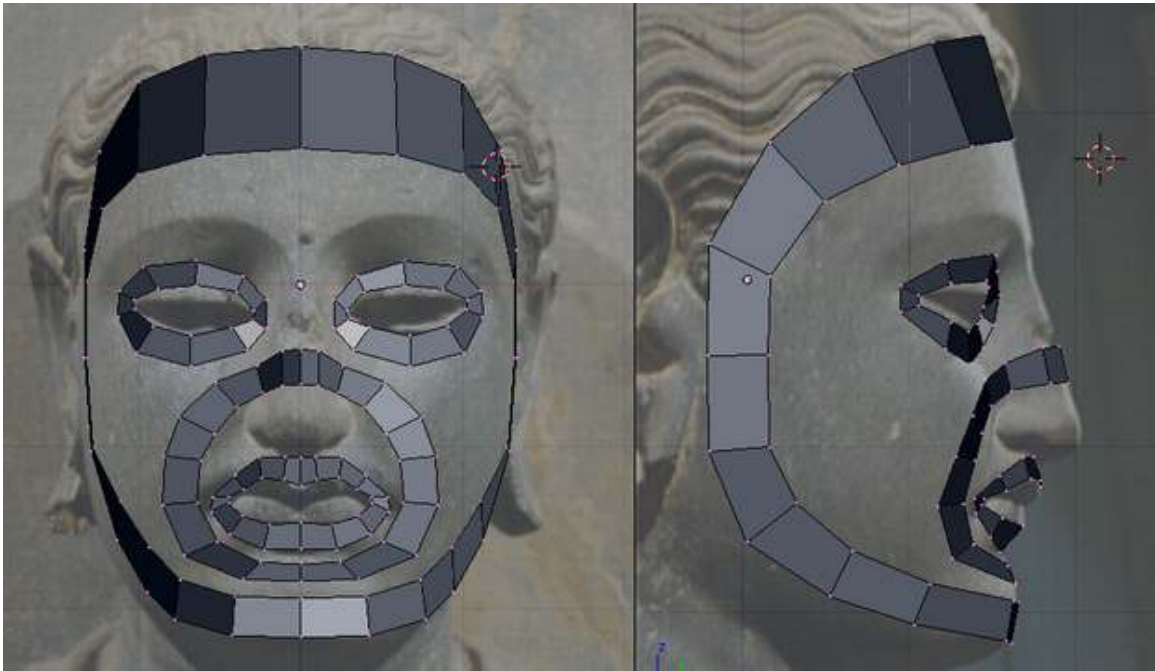


Figure 1541

Another good approach is to start with the most important part of the subject. For the face, this is the eyes and the mouth. You could model these parts first and then proceed adding geometry from there.

p Starting with the mouth

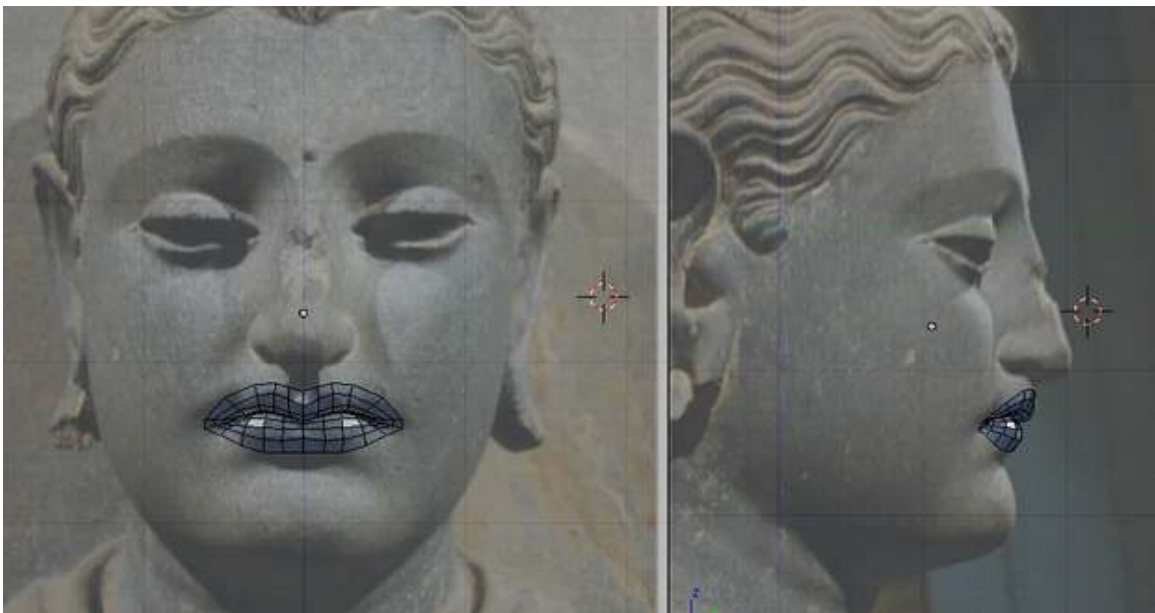


Figure 1542

As said, poly-by-poly modeling lets you free, so if you would like to start on other parts first, say the nose, then you could do so.

239.2.2. To Continue

p Continue Adding more geometry. Notice how the loops are placed



Figure 1543

Continuing from your beginnings is not as easy as connecting-the-dots as it first appears to be. Rather than proceeding in a connect the dots fashion, proceed by thinking of edge and face loops of your

model first then creating and adjusting geometry to fit them. Such view would ease decision on placing your geometry.

239.2.3. To Finish

p Rendered face after few adjustments



Figure 1544

When you find that you don't need to add more geometry, you can call it quits. But still, its a good practice to take another look of your model and make trial renders of it. There might be an awry

edge loop, geometries that mess the model, or triangles that you wanted out. Do some fixing and adjusting and when you finally have the model good, give yourself a pat on the back, OMG ITS A FACE YOU DID IT YAYAYAYAYA

240. Animation Notes and FAQ

240.1. Blender 3D: Animation Notes and FAQ

240.2. IMPORTANT

This page is under heavy construction. It will probably not be edited better until early May 2012. If someone sends me information on how to convert a OpenOffice odt file to a wikidoc format, then it may be done sooner. Until then this will be quite a work in progress.

240.3. Authors & Contributors

See [here](#)¹.

240.4. Introduction

I found it quite cumbersome to find all of the little quirks, problems, and tricks in multiple sources whenever I would forget something. Therefore, I compiled all of the things I found important—primarily with character animation. Almost all the information here is consolidated from many sources. Some sources still need to be cited and any help on this would be much appreciated. For that reason I did not intend to make this public but I think there is enough need for this type of information so here it is.

Just as a special aside, I made these notes to help myself with my animation project. Although I have organized them and posted them online for my and others benefit, I will not be dedicating any large amount of time to this website. All of my extra time needs to be allocated to the animation project. However, I do not mind suggestions and am willing to possible use them. I want to make sure that the viewer understands my time is very limited.

There may be many typos or confusing areas of organization. I tried to organize the mess of notes as best as possible but I am sure there is an even better way. Feel free to fix any aspect of this.

Any suggestions or comments feel free to e-mail me at [wiki \[at\] pagodaproductions.com](mailto:wiki@pagodaproductions.com).

¹ http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Authors_%26_Contributors

240.5. Todo List

See [here](#)².

240.6. Animation FAQ

240.7. Armatures

Q: I managed to create an acceptable walk cycle finally, now when I reopend the file to add additional animation, It keeps falling back to walk cycle, is there something i am doing wrong. Even after adding the LocRotScale key frame, it still ignores them, and goes back the corresponding frame for the walk. Uggggghhhh!!!

Q: What is the best approach for rigging a character in Blender?

Q: How do I attach a separate object to a bone?

Q: When you try to scale the root bone of the armature(which scales the mesh), the bones move out of the natural placement causing unwanted positioning and distortion. Is there any way around this?

Q: What is a pole vector?

Q: How do I know that my armature is correctly positioned in global space in edit, object, and pose mode with proper bone rolls and axis pointing in proper directions? What are the best methods for building? And what axis is up Y or Z?

240.8. Hotkeys

Ctrl-C/Ctrl-V Trick One simple way to get the right name is to select the Lattice, go to F9, Link and Materials panel and where it says Ob:Lattice or Ob:Lattice.001 etc., move the mouse over this field and press Ctrl-C (don't click on it, just hover over it). This copies the Lattice name. Then select your object, go to the Lattice Modifier panel, hover the mouse over the Ob: field and press Ctrl-V to paste the name in. Now it should stay there and your Lattice should work in Edit Mode.

H hides bones

Alt+H shows all bones

Shift+H hides everything except the chain

Ctrl+I [Pose Mode] will automatically add an IK constraint

M [Pose Mode] to move a bone to a bone-layer, select it in POSE mode then press "m" and click on the desired layer button. When the bone is selected, you can see which bone layer it is on in the Editing buttons / Selected Bones panel. Whichever of the 16 layer buttons is enabled is the layer the bone will be on.

² http://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/ToDo

Shift+M [Pose Mode] You can use the hotkey SH-M to display a floating bone-layer panel in the 3d viewport. Select the armature, change to POSE mode, then change to the Buttons/Editing panel (F9). Under the Armature-Panel, Display Options SH-LMB the 8th button under the Display Options Bone-Layer buttons to enable it. Check out [this](#)³ by BlenderArtists.com.

Alt-N normalize the bone orientation This can fix problems sometimes.

Alt+P clears parent

Alt+S [Pose Mode] in edit mode allows b-bones shape to scale visual changes without affecting the bones.

W [Edit Mode] subdivides bones

W [Pose Mode] will calculate paths to see how a bones movement is mapped out along a path or If you've named your bones correctly (with .L and .R at the end of the name) , you can flip the mirrored bone name by pressing "W" and "flip name" . A relevant post on BlenderArtsists.com is available [here](#)⁴.

240.9. Approaches, Techniques, Tips and Tricks

Constraining IK or FK In the Armature Bones Panel is where you can limit the movement of the bone for IK and Limit Rotation Constraint for FK

Customizing Frame Skipping Steps(F10 playback) allows for a customizable way of skipping frame in increments.

Copying Constraints in pose mode 1. Select all bone(s) you wish to copy the constraints to, from a particular bone 2. Now, select the bone you wish to copy constraints from 3. Ctrl C --> Constraints (All)

You can do CTRL-CKEY to copy stuff from a bone to bones. The options are location, rotation, scale and constraint. Constraint is very handy when you want to copy a constraint to other bone. The way it works is easy. The WKEY menu get some neat options too: Select constraint target: Will select the target of the bone's constraint currently selected. Flip name: Yep, you can flip name in Posemode too. Calculate/Clear path: This is a visual way to see the action linked to your armature. You can select just some bones and ask Blender to show you the paths of the bones.

It's possible to copy constraints from one object/bone to a bunch of objects/bones. A useful thing to know when doing a repetitive task like rigging all the fingers of a hand. Just select all bones/objects that you want to give a copy of the constraint, and then select the bone/object containing the constraints. Press CTRL-CKEY in 3DView, and select Object Constraints from the popup menu. The idea behind this is to copy the constraints of the active object to the selection.

Rigging Q: What is the best approach for rigging a character in Blender?

3 <http://blenderartists.org/forum/showthread.php?t=88090&highlight=hotkey+bone+layer>

4 <http://blenderartists.org/forum/showthread.php?t=93705&highlight=mirrored+bones>

A: Apollux Honestly, I don't think that there is a golden set-up that will solve all your problems. For example, when I design a character rig I need to know beforehand what type of actions he will perform.

If you want general pointers, I guess this would do: Always set your rigs in a half relaxed fashion.. totally relax your hand for a few minutes and see the position of the palm and fingers, that is position you should model and rig-them on the 3D world. Same goes for mouth, legs, arms, etc. etc. etc.

When in doubt over IK or FK always choose IK, since it is quite easy to switch a IK rig into FK when needed and then back to IK.

Bone's axis orientation can't be an afterthought!! It is no coincidence that every self-respecting rigging tutorial for blender mention it... And don't trust blindly on the automatic axis orientation fix command (Ctrl-N), since it can introduce some odd solutions on complex armatures. Always check each bone by hand, even if Blender says they are fixed.

Ultimate rig design is not the in-and-all of character animation. Sometimes I would spent crazy amounts of time preparing a rig for a special squence, and in the end realize that if I had used a simple FK rig tricked by hand during animation I would get the same results in half of the time and half the frustration. Not because it can be done means that it must be done. BlenderArtists post here⁵.

Item Pickup A good use of it is to ask a character to pick up something. By having a bone or empty for each side of the relationship (hand <-> glass), as the hand approaches the glass, you can align the two empties and fire the constraint up (1.00) to stick them together. You add another child-bone in the middle of the hand to tell where the glass will be. Thus moving the hand will move the glass. On the side of the glass just add an empty and make it parent of the glass. Add a copy location to the empty pointing to the bone in the hand we just did. There you go. Of course when the hand rotates the glass will not. For that you will need to add a Copy Rotation Constraint. Before Blender 2.40, the above method was a good way of faking parent relationship without rotation. But now we have the hinge option which does the same.

240.10. Problems and Questions

Separate Object/Bone Attachment **Q:** How do I attach a separate object to a bone?

A: To do that simply Ctrl-Tab into Pose Mode with the armature selected then select the object/mesh you want to parent to a bone then shift-select the bone you want to be the parent and Ctrl-P>"MakeParent to">Bone .

Armature Scaling Problems **Q:** I have been trying to scale my character which has many similarities to the mancandy and ludwig rigs. However, when you try to scale the root bone of the armature(which scales the mesh), the bones move out of the natural placement causing unwanted positioning and distortion. Both the mancandy and ludwig rigs have this same problem. Is there any way around this?

A: Why are you trying to scale the root bone (to scale the whole model)? Are you doing this in pose mode or object mode? If you are trying to scale the whole model try it in Object Mode. Also check if you are scaling around the cursor instead of median point

5 <http://blenderartists.org/forum/showthread.php?t=18480>

Q: What is a pole vector?

A: Vertex Pusher 2) Pole vector is actually a term used in Maya. It refers to the direction a bone is set to point towards in Maya rigging. For example, the direction you want the knees to point towards can be set using pole vectors. Apparently, Blender should be getting this feature soon . Right now you have to add target bones so that the knees will only bend one way (see Ryan Dale's BSoD tutorial), but with the option to use pole vectors you won't need to do this in the future .

BlenderArtists post about bones⁶.

Bone Orientation Q: How do I know that my armature is correctly positioned in global space in edit, object, and pose mode with proper bone rolls and axis pointing in proper directions? What are the best methods for building? And what axis is up Y or Z?

A: 1. Build up all the bones, as you did before. (Every foot, arm, etc. should be a bit bent, and drawn from the proper view.)

2. Select all these armatures, and with Ctrl-J join them into one armature. (Or with parent/child connection, as is necessary.)

3. Select this armature, go to "Editing" (F9), and then turn on "draw axes". Then you can see the axes of the bones. You will see, that the axes are different. Some bones has the Z-axis up, some has the Z-axis to the left, etc.

4. First you should tell Blender that this is the original position of your skeleton/model by pushing Ctrl-A. This is applying rotation. This means, that now all the rotations of the bones will be "0, 0, 0", so if you push later Alt-R (clear rotation), you will get this position. (Turn on numerical menu with N, to see what happens when you make this apply rotation command.)

5. Select the whole skeleton/model again. Go to edit mode. Then push A, to select all the bones. (They should be yellow now.) Then hit Ctrl-N, and you will see all the axes will be changed to stay on the same way.

6. That is, your skeleton now is ready to define the IK solvers, and then the skinning, and finally the animation.

<http://blenderartists.org/forum/showthread.php?t=25653>

A: Vertex Pusher 4) Yes you should always add in top view[numpad 7]. An armature bone always has its axes set (Y is always up X side to side and Z depth) and you cannot change this like you can with mesh objects, so it seemingly won't matter which view you add a bone while in Edit Mode but in Object Mode your armature object will have a 90 degree rotation around the X axis (if you add it in front view) which can and usually does create problems down the road . If for some reason you want to add bones in front view simply start by adding the first armature bone in top view ad then switch to front view to add additional bones . You can just delete the first bone or rotate it while in Edit Mode .

Blender's default views have the Z as the "up" axis while the armature system is coded with the Y as the "up" (like most 3D apps)

When you rotate a bone in the front view, when the bone is aligned to that view and the Z is the up axis, you will give a roll value the bone because it is rotating perpendicular to its coded up axis ...

⁶ <http://blenderartists.org/forum/showthread.php?t=110721>

this can cause you problems in some situations when you want a "neutral" roll value in Pose Mode... in most cases you'll want to recalculate the roll value with Ctrl-N (which will remove the visible rotation and insert a value in the "Roll:" field in the Transform Properties subwindow) and give it a non zero roll value in Edit Mode so that in Pose Mode it is properly aligned and rotated for poses and constraints ...

You won't get that behavior if you rotate in either the top or side views if the bone is aligned to the view ...

It's just that if you have the Y axis (the axis the bone rolls on) perpendicular to the Global Y axis then you will have to recalculate the bone roll angles to get it aligned properly for pose space .

By "bone is aligned to the view" I mean what it is like when you first add a bone in a preset view, i.e. head at bottom and tip up at top in Edit Mode with no rotation applied .

<http://blenderartists.org/forum/showthread.php?t=110721>

A: Fligh If you look at your left hand, palm down, from above, and assume for this exercise that all four fingers have their X-Axis pointing towards the thumb and have a Roll Angle of zero (same as if you created an Armature in Top View), but the thumb would need a Roll Angle of about 90. So if you rotate the first digit of the fingers around their Local X (R-XX) they would go down perpendicular to the back of the hand, but if you rotate the thumb (Roll Angle 90 and again around the Local X) it would go Left, towards the fingers.

Ahh.... but who bothers to pose bones like that? You just grab the IK Target and move it!. Well the Transform Matrix is divided into four chunks (I think? I'm not sure of this but it helps to think of it [or look at it] this way and if you actually test it in Blender it works), Bone Space (the structure of the bone including Roll Angle), Armature Space (the structure of the Armature = the sum of all BoneSpace), Pose Space (transforms relative to Rest Position = recorded changes in BoneSpace within ArmatureSpace) and World Space (the sum of all the above relative to the (transforms of) the Armature Object.

<http://blenderartists.org/forum/showthread.php?t=102209>

A: Fligh Anyhow, In Blender, In Top View (always add armatures in Top view) the Z-Axis is Facing you and Y is Up. So when you add the first Bone it lies along the Y-Axis. You cannot change that parallel-to-Y and any future Bones you extrude will use that bones axis as reference. If you add an Armature in Front view the bone's (local) Z-axis will point to the Global -Y (Y-Negative) and can lead to problems later on.

<http://blenderartists.org/forum/showthread.php?t=102609&highlight=bone+orientation>

A: Vertex Pusher Well I guess I may have a little something to contribute to this ... but I have been trying to learn the armature system in depth for the past few months and this particular issue came up with a complicated rig I was working with ... and I basically "discovered" that you really shouldn't have any other axis other than the Y as the "up" axis . And precisely because the bone roll angle issue .

It seems that despite how the view ports are configured (with the Z as the "up" axis) the default "up" axis is the Y ... for everything ... This is a bit annoying but if you have, like me, configured Blender to start with the front view open and have the Transform Properties window always open, you will see that everything when created in the front view is actually rotated 90 degrees along the X axis once you tab into Object Mode and that you need to Ctrl-A all objects so created (I guess Blender

"out of the box" has the top view open ? ... it's been a while so I can't remember ...) . This isn't all that important for most things but for the armature system this does create a problem - bone roll values that you don't want .

Just as a simple test : Open up Blender and add an armature in the front view . Grab the tip and move it to a corner of the grid so that it is angled 45 degrees . Tab into Object mode . Open up the Transform Properties window . You will notice that your armature is rotated 90 degrees along the X axis . So ... Ctrl-A the armature to have it's "up" axis as Z ... And suddenly you will see the bone rotate along its roll axis . Tab back into Edit Mode, select the entire bone and Ctrl-N to recalculate the bone roll angles ... the bone will rotate back to the way it was ... but with a change in its roll value (-45 or 45 depending on which way it is angled) ... Now repeat the above but this time create the armature in top view ... And you will realize that despite being angle at 45 degrees to the view like above no roll value ... not even after recalculating the bone roll angles (Ctrl-N) ...

Now I am assuming that this occurs because the Y axis is the one used to calculate the bone's roll ... and that if you have anything other than the Y as the "up" axis the roll values get screwed up because now the roll axis (Y) is tangent to the up axis instead of being the up axis . Just turn on "Draw Axes" and look at what the "up" axis is for an individual bone is ... the Y .

And just another bit of proof ... The really great new "limit" constraints (loc,rot and scale) added in the last release ... Try and apply the limit rotation constraint to the first armature (the one with Z up and rotation applied) . Now turn LimitX on with no min or max values and do not turn on the "local" co-ordinate space ... it will fall flat along the Y axis as if created in the top view ... This is occurring because despite the fact that you have applied the 90 degree rotation along the X axis to the object, the constraint is enforcing the global up axis ... which is ... the Y .

Now the only work around that I could come up with was to create an empty and parent a Y up armature to it then rotate the empty 90 degrees to be able to use the Z up view ports and also making sure the mesh also had the proper orientation as the armature . This prevents me from accidentally applying the rotation to the armature and I can use the numbpad hotkeys to navigate the views ... Else you could just clear the roll values manually if you insist on having the Z as the up axis ... though you won't be able to get rid of the rotation that is caused by being tangent to the Y axis without odd roll values ...

Sorry for the long winded response, but this has been bugging me for a while ... Hopefully with the refactoring for v2.5 later this year there will be an option as to which axis will be the "up" axis with regards to the various views ..

<http://blenderartists.org/forum/showthread.php?t=93578>

240.11. General Armature Information

Armature Object is like any other object type: It has a center, a position, a rotation and a scale factor. It can be edited. It can be linked to other scenes, and the same armature data can be reused on multiple objects. All animation you do in object mode is only working on the object, not the armature's contents like bones. Link and Materials panel: The AR: field let you rename your armature Datablock. The dropdown is a quick way to select which Armature datablock you want to connect to this armature. You can keep more than one version for the same character. Useful when you have a special move to achieve in a shot, you can turn on an armature for a special purpose. The F button

is an option to assign a Fake user to the Armature. Again if you have more than one armature for your character, it's a good idea to turn the Fake on, because if your armature datablock is not used (linked) it's not going to be saved in your .blend files. You can always do batch Fake-assignment of armatures by opening the Datablock browser (SHIFT-F4KEY), go in Armature datablock, select all the armatures you want to keep, and Press the FKEY. The OB: field is just to Rename your armature Object to something more cool and useful than Armature... Armature.001...

Delay Deform: This was useful before as the old system was very slow. What it does is when you do a manipulation to the rig, it waits until you finish to update the view. Can still be useful though.

Weight: This specifies how strongly this bone will influence the geometry around it, relative to the other bones. If two bones crossing each other, both with envelope influence, have the same weight (like 1:1) they will influence the surrounding geometry equally. But if you set one to 0.5, the geometry will be affected more significantly by the other one, with weight 1. For example, in this image, 2 bones using envelope influence try to move the same geometry. The 2 on the left have the same weight, you can see the geometry didn't move. On the right, one of the bones has 0.5 so the bone with weight 1 is winning the tug-of-war!: Deform: This lets you say if you want the bone to deform the geometry at all. Switching it off is like setting the weight to 0, except it's faster this way. Useful when using a bone as a target or a controller, i.e. a bone you just want to use to control other bones, but not the geometry itself. Mult: to deform geometry you can use vertex group and/or Envelope. The ability to mix both of these methods is handy for using one to tweak the other. For example, you might use envelope everywhere but tweak difficult places manually with vertex group. We'll discuss this in more detail later on. There are two number fields to better tweak the effect of B-Bones. The in/out is used to tell the scale of the virtual handle of the bezier curve. In is the Root of the bone, and Out is the Tip. The bigger the value, the bigger the effect of rotation. You can do ALT-SKEY on one or more bones while in Envelope display mode to tweak the envelope size in real time while animating. Useful when for example you move the hand and some part of the character isn't in the influence zone; the result will be that some vertices will stay behind.

Tip: Bake envelope to vertex groups The workflow is very simple. When you are done with the envelope's tweaking and you have gotten the best out of it, delete the Armature modifier and parent the mesh to the armature(CTRL-PKEY). Parent it to "armature" when asked and "Create From Closest Bones". Do ALT-PKEY and redo the Armature modifier. Now all the envelope influence are converted to Vertex Groups. This way you can further tweak influence zone using Weight paint. More info in the following pages. You can edit this white zone in Editmode or posemode by going in Envelope display mode, selecting bones and using SKEY or ALT-SKEY The Mult option will tell Blender to multiply the weight it get from envelope (let say 0.7) with the weight you painted in weight paint (let say 0.5). The result will be $0.5 * 0.7 = 0.35$ so in fact you just tweaked the envelope influence to 0.3 when it was at 0.7. If you don't want vertices to be part of the zone, you can always paint it with 0, as $0 * (\text{something})$ will always give 0. This way you can give custom shape to your envelope. More on weight paint on next page.

Deform with only things weight painted Deform + Multiply with only things that are weight painted within the envelope radius

Process: Make bones. Make good envelope transformations(including some overlap). Alt+P to make virtual parent(from nearest bones)this will already weight paint parts based on envelope influence

"All faces" tells Blender if you want to paint on all faces in the mesh or just the visible one. "Vertex Dist" tell blender to use vertex distance instead of faces. When active, the painting will only check if the vertex is in the brush, then apply a weight value. If it's off, all vertice part of the faces in the

brush will receive weight value. Turning on Vertex Dist can give good results when you have a lot of polys in your mesh. Constraints are calculated from first to last. So if you have two Constraints working on the same channel, let say Location, The last one will most probably win the chance to move the object. But... Most of the constraints have influence slider to tell how much it influence on the stack. If the last constraint have an influence of 0.5 it will mix the result with the one before. Most of the time this little constraint is useful to stick objects to one another. By playing with the Influence you can tell when it will work, when it will remain motionless.

240.12. Building Armatures

Ludwig's Stretchy Spine In the spine Ludwig model keep in mind the following things. The SpineStretch.null is necessary. It must be set to Hinge and Deform to keep the bones from scaling. IKSpine must have Deform turned off(none selected). Spine 4 or the last bone child in the chain(closest to the neck) must have Deform turned off(none selected). Scaling problem is usually tied(maybe always) tied to an improper child/parent relationship to a bone that is involved with stretching.

Created a mirrored copy without using X-Axis Mirror select the bones that you want to mirror , put the 3D cursor at the center : And make shift + d , and directly , S , X , -1 ...

Other Information You can extrude a new bone from the selection using EKEY. This will create a bone connected to the original one, meaning the Root of the new bone will follow the Tip of the original one. You can also CTRL-LMB to extrude a new bone. It will extrude to where you clicked.

Alternatively, you can connect two existing bones by selecting them one after the other and pressing CTRL-PKEY. You can then choose either 'Connected' (the child bone - the one you selected second - will automatically be moved so that it touches the parent) or 'Keep offset'. aa Using the WKEY menu, You can subdivide your bone or flip the name of the bone between Left-Right (See Naming convention below). You can delete the bone with XKEY You can select a chain of bones (connected together) using LKEY, when you hover your mouse over a bone. In many cases, rigs are symmetrical and can be mirrored in half. In these cases, it is helpful to use a left-right naming convention. This is not only useful for your own sake, but it gives Blender a hint that there is a pair of equivalent bones, and enables the use of some very cool tools that will save you some significant work. It's helpful to name your bones with something useful telling you what it's there for, such as leg, arm, finger, back, foot, etc. If you get a bone that has a copy on the other side, however, like the arm (you have 2 arms right?), then the convention is to call them arm.Left and arm.Right. Other alternatives are also possible, like _L, _LEFT, _left, .L, and .Left. Anyway, when you rig try to keep this left-right thing as accurate as possible; it will pay off later on. You can copy a bone named blah.L and flip it over using WKEY --> flip name. So the bone will be blah.L.001 after you copy it, and flipping the name will give you blah.R. Blender handily detects if the .001 version already exists, and increments the number for you.

240.13. Useful Rigs

Bassam's(slikdigit) Mancandy 1.0 <http://freefactory.org/posts/candy-for-everyone>

Bassam's(slikdigit) Mancandy 2.0 One of the most complex rig I have available for Blender. It has lattice based stretching. <http://freefactory.org/posts/candy-for-everyone>

Calvin's 05, 2017 This is an alternate foot roll method to Ludwig's rig.

Calvin's handy02

Clean3D's Mouserig <http://blenderartists.org/forum/showthread.php?t=79305>

Cognis's AnimTest Interesting use of empty's as a bone.

Daniel Martinez Lara's 3-plender leg This is a complex leg rig. It has many IPO driven bones. http://www.daniel3d.com/pepeland/misc/3dstuff/blender/rig/3-plender_leg_v0-1.zip

Jason Pierce's(sketchy) Ludwig <http://jasonpierce.animadillo.com/resource/ludwig/ludwig.html>

Kugyelka's Walkcycle

Martin Georgiev's(Animarto) Walk_anim

Master Yoda

Michael Thoenes's Suzanne Rigged <http://creationanimation.com/sites/thoenes/animation/3dtutorials.htm>

Nathaniel Shaw's Generi

Noodlesgc's (Mike Roberts) W1r3z <http://www.esnips.com/doc/276d8bd1-0070-4ee4-bd8c-e787b507d6a9/W1r3z.blend>

Nozzy's Driven hand <http://blenderartists.org/forum/showthread.php?t=18754>

Nozzy's Skinny Guy <http://blenderartists.org/forum/showthread?t=30994>

Otothecleaner's Flor

Virgilio's Otto 1.6 Interesting constraint to keep the feet on the floor. http://uploader.polorix.net/files/99/otto_v1.6.zip

Rbackman's Bounce9

Rbackman's Zeyne

Simple's Character

Tugkan's Sharlo

Woodman5k's Bunny <http://web.pdx.edu/~wlf/Bunny.rar>

Yagapayanata's Rig

Yagapayanata's Rig(2, 4, 6 legs)

241. Customization

241.1. Changing the Theme

Like many other programs with a graphical user interface, Blender allows you to customize or modify many aspects of its appearance. A collection of such customizations is called a **theme**.

Theme controls are found in the User Preferences window. Click **LMB** on the "Themes" button to activate a drop-down menu which can be used to select different themes. Below this is an "Add" button that can be used to create new themes. If any themes have been added, a "Delete" button will appear, along with other controls to adjust the current theme.

This book presents screen shots using the default theme. If you are new to Blender, you should continue using the default theme as you progress through the book, to avoid confusion.

242. Mist - Make Objects Opaque

How Blender's Mist works

If you render Objects within Blender's Mist, they mix up with the Background-Color,

because Blender decreases their Alpha-Value by increasing Mist-Intensity and distance from the Camera.

The Problem

This does a well job if your scene has a constantly-colored Background, but

in case of a real Background or World-Texture it looks not convincing.

All objects are looking like ghosts.

The Solution explained briefly

This page will show you two ways to get rid of that Problem.

The major thought is to mix Objects with an opaque color, instead of making them transparent.

The more Transparency, the more of the opaque color will be added.

Because Blender delivers a (dynamic) Alpha-Value with Mist enabled,

we can use this value as a factor for increase or decrease of mixing the objects

with a certain color (or even to enhance or lower the contrast of materials, etc.)



Figure 1545 Here's a render result with mist and the "Mixing-Technique". You see, that you don't see shining the Background through.

The first solution goes with the help of an external compositing program

and won't be explained in depth as we want to focus on how to solve

this issue within blender since it is open source.

Solution 1: Make use of an external composite software.

Just render your Background and your Objects separately with Alpha-Channels.

Then you import these files into a composite program on two layers.

(One for the Objects and one for the Background).

Between these layers you set a new "Color-Layer" with a solid greyish-blue color

(because this fits the color of the atmosphere) and create a clipping mask.

A "clipping mask" means that the "Color-Layer" is only visible where the objects of the top layer are.

And where there's no object you can see the Background.

As the object is partially transparent (because of Blender's Mist),

you will see the greyish-blue color shining through.

The more the object is away, the more transparent it is,

and the more the color-Layer will shine through the Objects.

Solution 2: Make use of Blender's Composite-Nodes.

Since Solution 1 seems not to be a big deal as we cannot afford expensive packages, we are also able

to achieve this "Mix-Technique" by Blender's built-in Composite-Nodes.

To continue reading it would be good, if you know a little about Blender's Compositing-Nodes.

Now, here's the interesting part of the tutorial, the Graph of the Nodes in the Node-Editor:

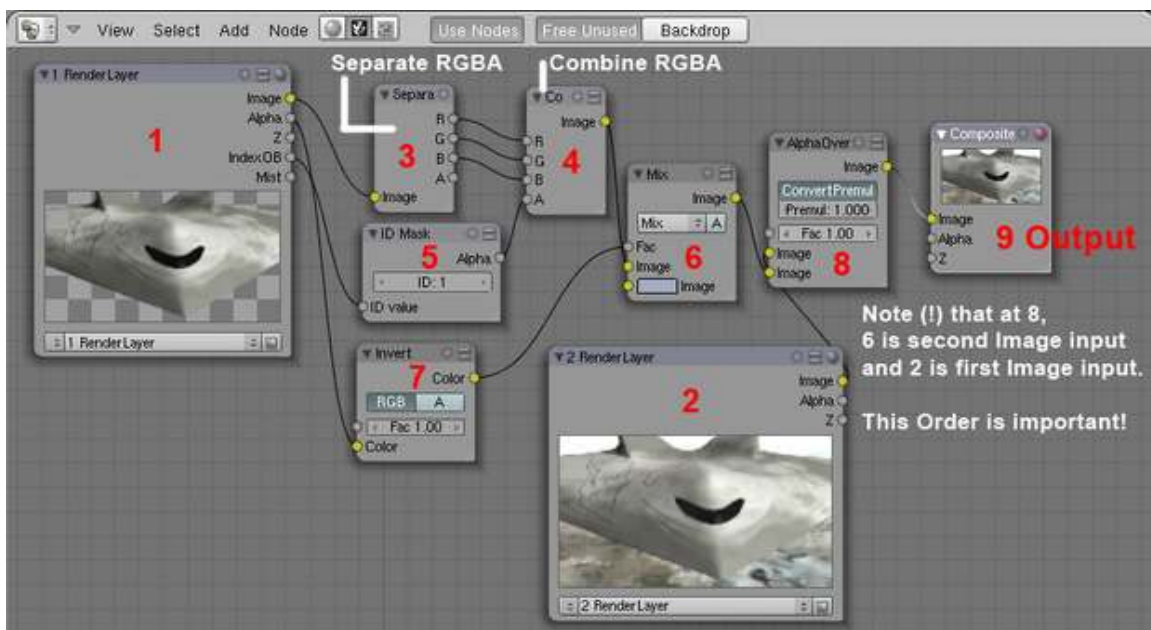


Figure 1546

Description:

Step 1 Render Layer1

In the Render Settings Tab you assign the Objects which you don't want to be transparent to Render Layer1.

See <http://wiki.blender.org/index.php/Doc:Manual/Render/Layers>¹ for how to deal with RenderLayers.

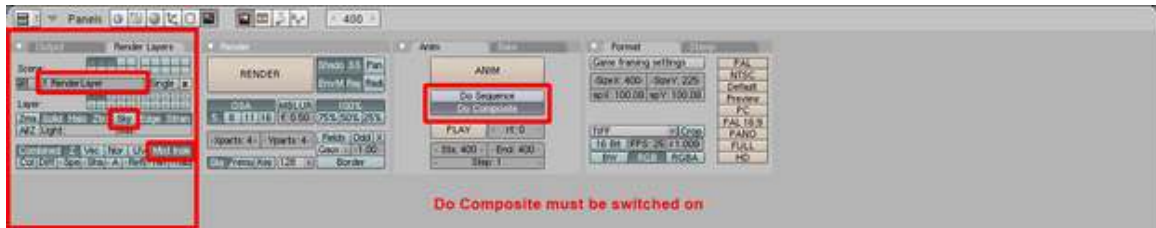


Figure 1547

It is important that you turn off "Sky" and that you turn on "IndexPass". (The Option "Mist" delivers the same Value as the Alpha-Output in the Node RenderLayer1)

"IndexPass" will be explained in Step 5.

Step 2 Render Layer2

This Layer renders only the Background, the Sky. In my case I had to enable other elements also, because otherwise the groundplane won't render.

I don't know why, but this doesn't matter in our case.

Step 3 Separating RGB+Alpha

The "Image output" of the Node RenderLayer1 delivers RGB+A, that's a little distracting,

because you can see an Alpha-Output also. Select "Add" -> "Color" -> "Separate RGBA" in the Node-Editor Window.

Step 4 Combining RGBA

Now the Separated RGB gets combined again to RGB, but this time with a new Alpha of Step 5.

¹ <http://wiki.blender.org/index.php/Doc:Manual/Render/Layers>

Step 5 Making the transparent Objects opaque

The good thing is, that Blender's Object-ID Alpha Value does ignore mist.

So we can add an ID-Mask Node to get the Alpha of the Object into the new Image.

Here you must set the ID of the Object to the same number as in the ID-Mask Node.

Select the Object and set its ID Pass-Index!

Beside is a screenshot where you set the Object ID.



Figure 1548 In the red marked section of the buttons window, you can set the ID of the object

Step 6 Mixing the Object with a Color

Here you add a Mix-Node to mix the Object with a "Horizon-Color".

The Color should be set to a bright greyish-blue as this is the color of the air inner the atmosphere.

Any color is possible, you may think of a dark red in a fire-hollow or of black if somewhere out in space.

Step 7 Setting the Factor of how much of the color should be added

Blender delivers an Alpha-Value to the Objects due to the Mist.

If an object is far away or if the Mist-Intensity is very high,

the Alpha-Value of the Object decreases, which means the Alpha-Value becomes more black.

And if the Object is close, it has accordingly high Alpha, which means getting white.

Black means a Value of R-G-B 0-0-0 and White a Value of R-G-B 1-1-1.

Since we want to use these values as a factor of how much to blend the Object

with our "Horizon-Color", we need to invert this Value.

(Close Objects = High Alpha = Factor almost 1 - but we need Blend Factor close to 0, because we want low mist as the Object is close)

So let's add an "Invert-Color-Node".

Get the Settings of the Node-Graph for this Node.

This Node inverts Black to white, dark-grey to bright grey, etc.

We now use this inverted alpha-value (or Mist-value)

as a factor for mixing the Horizon-Color with the Object.

Step 8 Layering the colored Objects over the Sky-Background

Now you add an Alpha-Over-Node which will layer our Objects over the Sky-Background.

Be aware to use the settings shown in the screenshot and that the order of the two Image-Inputs

of the Alpha-Over-Node is very important.

The second Input always gets over the first Input.

Step 9 Add an Output-Node

Finally you need an Output-Node and to enable "Do Composite" in the RenderSettings.

Step 10 Example

Here an Example, of how it worked out in my project:



Figure 1549 Close Distance - Almost no Mist



Figure 1550 Far Distance - Low Mist or atmospheric opacity



Figure 1551 Further Distance - (!)The Object is even scaled



Figure 1552 Farthermost Distance - In original size the Object would be so small that it isn't visible -only to make the Logic clear

Summary

Be sure, you got every step right - then all will work out properly.

What is almost of more importance, that you understood the logic behind this,

so that you learn about Blender's Compositing-Nodes and the 01-Logic ;)

Since "Mist" means something like "Bullsh**" in German I did not believe that it really

is like that in Blender - and it isn't! Blender is a quite nice and powerful tool.

And I hope that it'll ever be Open-Source.

243. Contributors

Edits	User
3	00Anme00 ¹
3	01mf02 ²
1	0o0o0ok ³
3	1983 ⁴
1	2DCube ⁵
2	3DSki ⁶
2	492star ⁷
2	4thought ⁸
2	5o ⁹
1	80063r ¹⁰
32	AM088 ¹¹
1	Abcdaa ¹²
2	Accurrent ¹³
8	Acsteitz ¹⁴
1	AdRiley ¹⁵
1	Adamp ¹⁶
1	AdemoS ¹⁷
2	Adhemar ¹⁸
29	Admijr ¹⁹
300	Adrignola ²⁰
1	AfroToad ²¹

1	http://en.wikibooks.org/w/index.php?title=User:00Anme00
2	http://en.wikibooks.org/w/index.php?title=User:01mf02
3	http://en.wikibooks.org/w/index.php?title=User:0o0o0ok
4	http://en.wikibooks.org/w/index.php?title=User:1983
5	http://en.wikibooks.org/w/index.php?title=User:2DCube
6	http://en.wikibooks.org/w/index.php?title=User:3DSki
7	http://en.wikibooks.org/w/index.php?title=User:492star
8	http://en.wikibooks.org/w/index.php?title=User:4thought
9	http://en.wikibooks.org/w/index.php?title=User:5o
10	http://en.wikibooks.org/w/index.php?title=User:80063r
11	http://en.wikibooks.org/w/index.php?title=User:AM088
12	http://en.wikibooks.org/w/index.php?title=User:Abcdaa
13	http://en.wikibooks.org/w/index.php?title=User:Accurrent
14	http://en.wikibooks.org/w/index.php?title=User:Acsteitz
15	http://en.wikibooks.org/w/index.php?title=User:AdRiley
16	http://en.wikibooks.org/w/index.php?title=User:Adamp
17	http://en.wikibooks.org/w/index.php?title=User:AdemoS
18	http://en.wikibooks.org/w/index.php?title=User:Adhemar
19	http://en.wikibooks.org/w/index.php?title=User:Admijr
20	http://en.wikibooks.org/w/index.php?title=User:Adrignola
21	http://en.wikibooks.org/w/index.php?title=User:AfroToad

3	Aggressor ²²
2	Aidan.Sullivan ²³
1	Akhram ²⁴
2	Alabandit ²⁵
5	Alchez ²⁶
4	Alex G ²⁷
1	Alex L33 ²⁸
1	Alexwill84 ²⁹
47	Allefant ³⁰
2	AllenZh ³¹
1	Alpha0 ³²
47	AmieKim ³³
6	Amigosg ³⁴
11	AndrewBuck ³⁵
5	Andrewtrefethen ³⁶
106	AndyD ³⁷
1	Aneeshnl ³⁸
1	Anonymous Dissident ³⁹
1	Antcodd ⁴⁰
1	Aphax ⁴¹
1	Arakade ⁴²
70	Arch dude ⁴³
1	Arch dude (usurped) ⁴⁴
8	Arcticfreeze83 ⁴⁵
9	Argav ⁴⁶

22	http://en.wikibooks.org/w/index.php?title=User:Aggressor
23	http://en.wikibooks.org/w/index.php?title=User:Aidan.Sullivan
24	http://en.wikibooks.org/w/index.php?title=User:Akhram
25	http://en.wikibooks.org/w/index.php?title=User:Alabandit
26	http://en.wikibooks.org/w/index.php?title=User:Alchez
27	http://en.wikibooks.org/w/index.php?title=User:Alex_G
28	http://en.wikibooks.org/w/index.php?title=User:Alex_L33
29	http://en.wikibooks.org/w/index.php?title=User:Alexwill84
30	http://en.wikibooks.org/w/index.php?title=User:Allefant
31	http://en.wikibooks.org/w/index.php?title=User:AllenZh
32	http://en.wikibooks.org/w/index.php?title=User:Alpha0
33	http://en.wikibooks.org/w/index.php?title=User:AmieKim
34	http://en.wikibooks.org/w/index.php?title=User:Amigosg
35	http://en.wikibooks.org/w/index.php?title=User:AndrewBuck
36	http://en.wikibooks.org/w/index.php?title=User:Andrewtrefethen
37	http://en.wikibooks.org/w/index.php?title=User:AndyD
38	http://en.wikibooks.org/w/index.php?title=User:Aneeshnl
39	http://en.wikibooks.org/w/index.php?title=User:Anonymous_Dissident
40	http://en.wikibooks.org/w/index.php?title=User:Antcodd
41	http://en.wikibooks.org/w/index.php?title=User:Aphax
42	http://en.wikibooks.org/w/index.php?title=User:Arakade
43	http://en.wikibooks.org/w/index.php?title=User:Arch_dude
44	http://en.wikibooks.org/w/index.php?title=User:Arch_dude_%28usurped%29
45	http://en.wikibooks.org/w/index.php?title=User:Arcticfreeze83
46	http://en.wikibooks.org/w/index.php?title=User:Argav

70	Argento ⁴⁷
2	Arru ⁴⁸
3	Arturotena ⁴⁹
6	Ascen ⁵⁰
1	Aschoeke ⁵¹
1	Ask ⁵²
1	Astharoth ⁵³
6	Astronouth7303 ⁵⁴
6	Atallcostsky ⁵⁵
3	Atarun ⁵⁶
3	Atimmer ⁵⁷
30	AttilaMH ⁵⁸
21	AverieZen ⁵⁹
82	Avicennasis ⁶⁰
11	Az1568 ⁶¹
15	BANZ111 ⁶²
4	Baadtaste ⁶³
1	Bach ⁶⁴
2	Bacon ⁶⁵
32	Badalia ⁶⁶
2	Bamacsguy ⁶⁷
2	Banzai ⁶⁸
1	BeBraw ⁶⁹
3	Beetlemaniac ⁷⁰
1	Begeun ⁷¹

47	http://en.wikibooks.org/w/index.php?title=User:Argento
48	http://en.wikibooks.org/w/index.php?title=User:Arru
49	http://en.wikibooks.org/w/index.php?title=User:Arturotena
50	http://en.wikibooks.org/w/index.php?title=User:Ascen
51	http://en.wikibooks.org/w/index.php?title=User:Aschoeke
52	http://en.wikibooks.org/w/index.php?title=User:Ask
53	http://en.wikibooks.org/w/index.php?title=User:Astharoth
54	http://en.wikibooks.org/w/index.php?title=User:Astronouth7303
55	http://en.wikibooks.org/w/index.php?title=User:Atallcostsky
56	http://en.wikibooks.org/w/index.php?title=User:Atarun
57	http://en.wikibooks.org/w/index.php?title=User:Atimmer
58	http://en.wikibooks.org/w/index.php?title=User:AttilaMH
59	http://en.wikibooks.org/w/index.php?title=User:AverieZen
60	http://en.wikibooks.org/w/index.php?title=User:Avicennasis
61	http://en.wikibooks.org/w/index.php?title=User:Az1568
62	http://en.wikibooks.org/w/index.php?title=User:BANZ111
63	http://en.wikibooks.org/w/index.php?title=User:Baadtaste
64	http://en.wikibooks.org/w/index.php?title=User:Bach
65	http://en.wikibooks.org/w/index.php?title=User:Bacon
66	http://en.wikibooks.org/w/index.php?title=User:Badalia
67	http://en.wikibooks.org/w/index.php?title=User:Bamacsguy
68	http://en.wikibooks.org/w/index.php?title=User:Banzai
69	http://en.wikibooks.org/w/index.php?title=User:BeBraw
70	http://en.wikibooks.org/w/index.php?title=User:Beetlemaniac
71	http://en.wikibooks.org/w/index.php?title=User:Begeun

- 1 Beginner⁷²
- 1 Behemoth14⁷³
- 2 Bekenn⁷⁴
- 2 Benwulf⁷⁵
- 35 Beuc⁷⁶
- 9 Bgks⁷⁷
- 1 Bijee⁷⁸
- 2 Billinghamurst⁷⁹
- 1 Billymac00⁸⁰
- 1 BimBot⁸¹
- 2 Biomeksensei⁸²
- 2 Bk314159⁸³
- 3 BlenderDocs⁸⁴
- 1 Blendergeek⁸⁵
- 1 Blenderizer⁸⁶
- 1 Blenderman⁸⁷
- 1 Blondandy⁸⁸
- 1 Blueskies⁸⁹
- 1 Bluincher⁹⁰
- 2 Bmarnz⁹¹
- 4 Bmud⁹²
- 1 Bobj⁹³
- 2 Bocercus⁹⁴
- 71 Bonapon⁹⁵
- 1 Booner⁹⁶

-
- 72 <http://en.wikibooks.org/w/index.php?title=User:Beginner>
 - 73 <http://en.wikibooks.org/w/index.php?title=User:Behemoth14>
 - 74 <http://en.wikibooks.org/w/index.php?title=User:Bekenn>
 - 75 <http://en.wikibooks.org/w/index.php?title=User:Benwulf>
 - 76 <http://en.wikibooks.org/w/index.php?title=User:Beuc>
 - 77 <http://en.wikibooks.org/w/index.php?title=User:Bgks>
 - 78 <http://en.wikibooks.org/w/index.php?title=User:Bijee>
 - 79 <http://en.wikibooks.org/w/index.php?title=User:Billinghamurst>
 - 80 <http://en.wikibooks.org/w/index.php?title=User:Billymac00>
 - 81 <http://en.wikibooks.org/w/index.php?title=User:BimBot>
 - 82 <http://en.wikibooks.org/w/index.php?title=User:Biomeksensei>
 - 83 <http://en.wikibooks.org/w/index.php?title=User:Bk314159>
 - 84 <http://en.wikibooks.org/w/index.php?title=User:BlenderDocs>
 - 85 <http://en.wikibooks.org/w/index.php?title=User:Blendergeek>
 - 86 <http://en.wikibooks.org/w/index.php?title=User:Blenderizer>
 - 87 <http://en.wikibooks.org/w/index.php?title=User:Blenderman>
 - 88 <http://en.wikibooks.org/w/index.php?title=User:Blondandy>
 - 89 <http://en.wikibooks.org/w/index.php?title=User:Blueskies>
 - 90 <http://en.wikibooks.org/w/index.php?title=User:Bluincher>
 - 91 <http://en.wikibooks.org/w/index.php?title=User:Bmarnz>
 - 92 <http://en.wikibooks.org/w/index.php?title=User:Bmud>
 - 93 <http://en.wikibooks.org/w/index.php?title=User:Bobj>
 - 94 <http://en.wikibooks.org/w/index.php?title=User:Bocercus>
 - 95 <http://en.wikibooks.org/w/index.php?title=User:Bonapon>
 - 96 <http://en.wikibooks.org/w/index.php?title=User:Booner>

36	Bootaleg ⁹⁷
3	Borisblue ⁹⁸
12	Boyage ⁹⁹
1	Boywonder ¹⁰⁰
9	Brandished ¹⁰¹
4	Brian 252 ¹⁰²
9	Brinman ¹⁰³
2	Bronzed bison ¹⁰⁴
2	Bruce89 ¹⁰⁵
5	Bruyninc ¹⁰⁶
3	Brylie ¹⁰⁷
5	Bturnip ¹⁰⁸
1	Buckbeak ¹⁰⁹
194	Bullercruz1 ¹¹⁰
3	Bunyk ¹¹¹
2	Burpen ¹¹²
1	CFrey ¹¹³
9	Cale ¹¹⁴
1	CamStone ¹¹⁵
3	Camel ¹¹⁶
22	Camp ¹¹⁷
3	CaptBobo ¹¹⁸
2	Carpetsmoker ¹¹⁹
1	CarsracBot ¹²⁰
1	Casimir ¹²¹

-
- 97 <http://en.wikibooks.org/w/index.php?title=User:Bootaleg>
- 98 <http://en.wikibooks.org/w/index.php?title=User:Borisblue>
- 99 <http://en.wikibooks.org/w/index.php?title=User:Boyage>
- 100 <http://en.wikibooks.org/w/index.php?title=User:Boywonder>
- 101 <http://en.wikibooks.org/w/index.php?title=User:Brandished>
- 102 http://en.wikibooks.org/w/index.php?title=User:Brian_252
- 103 <http://en.wikibooks.org/w/index.php?title=User:Brinman>
- 104 http://en.wikibooks.org/w/index.php?title=User:Bronzed_bison
- 105 <http://en.wikibooks.org/w/index.php?title=User:Bruce89>
- 106 <http://en.wikibooks.org/w/index.php?title=User:Bruyninc>
- 107 <http://en.wikibooks.org/w/index.php?title=User:Brylie>
- 108 <http://en.wikibooks.org/w/index.php?title=User:Bturnip>
- 109 <http://en.wikibooks.org/w/index.php?title=User:Buckbeak>
- 110 <http://en.wikibooks.org/w/index.php?title=User:Bullercruz1>
- 111 <http://en.wikibooks.org/w/index.php?title=User:Bunyk>
- 112 <http://en.wikibooks.org/w/index.php?title=User:Burpen>
- 113 <http://en.wikibooks.org/w/index.php?title=User:CFrey>
- 114 <http://en.wikibooks.org/w/index.php?title=User:Cale>
- 115 <http://en.wikibooks.org/w/index.php?title=User:CamStone>
- 116 <http://en.wikibooks.org/w/index.php?title=User:Camel>
- 117 <http://en.wikibooks.org/w/index.php?title=User:Camp>
- 118 <http://en.wikibooks.org/w/index.php?title=User:CaptBobo>
- 119 <http://en.wikibooks.org/w/index.php?title=User:Carpetsmoker>
- 120 <http://en.wikibooks.org/w/index.php?title=User:CarsracBot>
- 121 <http://en.wikibooks.org/w/index.php?title=User:Casimir>

- 1 Casper da Costa-Luis¹²²
- 1 Cesarespcab¹²³
- 1 Chandhamama¹²⁴
- 1 CharlesLambert¹²⁵
- 17 Charybdis¹²⁶
- 11 Chelseafan528¹²⁷
- 8 Chinmay gautam¹²⁸
- 5 Chris0917¹²⁹
- 1 Chuckhoffmann¹³⁰
- 1 Chuey¹³¹
- 3 Chw333¹³²
- 1 Citizen428¹³³
- 2 Clemens¹³⁴
- 2 Cleobod¹³⁵
- 8 Clone Dad¹³⁶
- 11 CodeSnorter¹³⁷
- 1 Coffeeparis¹³⁸
- 1 Colincbn¹³⁹
- 15 CommonsDelinker¹⁴⁰
- 1 Comwizz2¹⁴¹
- 3 Corn Chowder¹⁴²
- 1 Cosworth¹⁴³
- 1 Cptlongshlong¹⁴⁴
- 2 CrAc¹⁴⁵
- 27 Crouch¹⁴⁶

-
- 122 http://en.wikibooks.org/w/index.php?title=User:Casper_da_Costa-Luis
 - 123 <http://en.wikibooks.org/w/index.php?title=User:Cesarespcab>
 - 124 <http://en.wikibooks.org/w/index.php?title=User:Chandhamama>
 - 125 <http://en.wikibooks.org/w/index.php?title=User:CharlesLambert>
 - 126 <http://en.wikibooks.org/w/index.php?title=User:Charybdis>
 - 127 <http://en.wikibooks.org/w/index.php?title=User:Chelseafan528>
 - 128 http://en.wikibooks.org/w/index.php?title=User:Chinmay_gautam
 - 129 <http://en.wikibooks.org/w/index.php?title=User:Chris0917>
 - 130 <http://en.wikibooks.org/w/index.php?title=User:Chuckhoffmann>
 - 131 <http://en.wikibooks.org/w/index.php?title=User:Chuey>
 - 132 <http://en.wikibooks.org/w/index.php?title=User:Chw333>
 - 133 <http://en.wikibooks.org/w/index.php?title=User:Citizen428>
 - 134 <http://en.wikibooks.org/w/index.php?title=User:Clemens>
 - 135 <http://en.wikibooks.org/w/index.php?title=User:Cleobod>
 - 136 http://en.wikibooks.org/w/index.php?title=User:Clone_Dad
 - 137 <http://en.wikibooks.org/w/index.php?title=User:CodeSnorter>
 - 138 <http://en.wikibooks.org/w/index.php?title=User:Coffeeparis>
 - 139 <http://en.wikibooks.org/w/index.php?title=User:Colincbn>
 - 140 <http://en.wikibooks.org/w/index.php?title=User:CommonsDelinker>
 - 141 <http://en.wikibooks.org/w/index.php?title=User:Comwizz2>
 - 142 http://en.wikibooks.org/w/index.php?title=User:Corn_Chowder
 - 143 <http://en.wikibooks.org/w/index.php?title=User:Cosworth>
 - 144 <http://en.wikibooks.org/w/index.php?title=User:Cptlongshlong>
 - 145 <http://en.wikibooks.org/w/index.php?title=User:CrAc>
 - 146 <http://en.wikibooks.org/w/index.php?title=User:Crouch>

- 1 Crystal Visions¹⁴⁷
- 2 Cspurrier¹⁴⁸
- 9 Cthames¹⁴⁹
- 1 CwadrupldijjitGoldilox¹⁵⁰
- 1 Cymru.lass¹⁵¹
- 7 DDD3x¹⁵²
- 1 DEAFCODE¹⁵³
- 1 DRNZ¹⁵⁴
- 4 Dagnode¹⁵⁵
- 1 Dagonite¹⁵⁶
- 2 Dalhamir¹⁵⁷
- 2 Dallas1278¹⁵⁸
- 4 Damian Yerrick¹⁵⁹
- 3 Dan-aka-jack¹⁶⁰
- 3 Dan13¹⁶¹
- 4 DanMan¹⁶²
- 1 Dandellion¹⁶³
- 1 Danhash¹⁶⁴
- 3 Danielsimlind¹⁶⁵
- 1 Danthemango¹⁶⁶
- 1 Dapoulter¹⁶⁷
- 3 DariusWiles¹⁶⁸
- 12 Darkfire¹⁶⁹
- 7 Darklama¹⁷⁰
- 1 Darknessknight¹⁷¹

-
- 147 http://en.wikibooks.org/w/index.php?title=User:Crystal_Visions
 - 148 <http://en.wikibooks.org/w/index.php?title=User:Cspurrier>
 - 149 <http://en.wikibooks.org/w/index.php?title=User:Cthames>
 - 150 <http://en.wikibooks.org/w/index.php?title=User:CwadrupldijjitGoldilox>
 - 151 <http://en.wikibooks.org/w/index.php?title=User:Cymru.lass>
 - 152 <http://en.wikibooks.org/w/index.php?title=User:DDD3x>
 - 153 <http://en.wikibooks.org/w/index.php?title=User:DEAFCODE>
 - 154 <http://en.wikibooks.org/w/index.php?title=User:DRNZ>
 - 155 <http://en.wikibooks.org/w/index.php?title=User:Dagnode>
 - 156 <http://en.wikibooks.org/w/index.php?title=User:Dagonite>
 - 157 <http://en.wikibooks.org/w/index.php?title=User:Dalhamir>
 - 158 <http://en.wikibooks.org/w/index.php?title=User:Dallas1278>
 - 159 http://en.wikibooks.org/w/index.php?title=User:Damian_Yerrick
 - 160 <http://en.wikibooks.org/w/index.php?title=User:Dan-aka-jack>
 - 161 <http://en.wikibooks.org/w/index.php?title=User:Dan13>
 - 162 <http://en.wikibooks.org/w/index.php?title=User:DanMan>
 - 163 <http://en.wikibooks.org/w/index.php?title=User:Dandellion>
 - 164 <http://en.wikibooks.org/w/index.php?title=User:Danhash>
 - 165 <http://en.wikibooks.org/w/index.php?title=User:Danielsimlind>
 - 166 <http://en.wikibooks.org/w/index.php?title=User:Danthemango>
 - 167 <http://en.wikibooks.org/w/index.php?title=User:Dapoulter>
 - 168 <http://en.wikibooks.org/w/index.php?title=User:DariusWiles>
 - 169 <http://en.wikibooks.org/w/index.php?title=User:Darkfire>
 - 170 <http://en.wikibooks.org/w/index.php?title=User:Darklama>
 - 171 <http://en.wikibooks.org/w/index.php?title=User:Darknessknight>

- 3 Darkonc¹⁷²
- 5 Davekeiser¹⁷³
- 1 David Cochran¹⁷⁴
- 10 David Osborn¹⁷⁵
- 5 DavidBaird¹⁷⁶
- 1 DavidCary¹⁷⁷
- 8 DavidJaquay¹⁷⁸
- 1 Davidfg¹⁷⁹
- 1 Davidwilliams¹⁸⁰
- 7 Daworm¹⁸¹
- 1 DbD¹⁸²
- 2 Dbproguy¹⁸³
- 1 Dbreece¹⁸⁴
- 14 DeltaSpeeds¹⁸⁵
- 2 Demoathon¹⁸⁶
- 1 Dennis G. Jerz¹⁸⁷
- 1 Der Künstler¹⁸⁸
- 22 Derbeth¹⁸⁹
- 1 Dessydes¹⁹⁰
- 1 Diblidabliduu¹⁹¹
- 1 Digitalartist¹⁹²
- 1 Digitalcircuit36939¹⁹³
- 1 Digivation¹⁹⁴
- 3 Dimzy Doodle¹⁹⁵
- 2 Dipingo¹⁹⁶

-
- 172 <http://en.wikibooks.org/w/index.php?title=User:Darkonc>
 - 173 <http://en.wikibooks.org/w/index.php?title=User:Davekeiser>
 - 174 http://en.wikibooks.org/w/index.php?title=User:David_Cochran
 - 175 http://en.wikibooks.org/w/index.php?title=User:David_Osborn
 - 176 <http://en.wikibooks.org/w/index.php?title=User:DavidBaird>
 - 177 <http://en.wikibooks.org/w/index.php?title=User:DavidCary>
 - 178 <http://en.wikibooks.org/w/index.php?title=User:DavidJaquay>
 - 179 <http://en.wikibooks.org/w/index.php?title=User:Davidfg>
 - 180 <http://en.wikibooks.org/w/index.php?title=User:Davidwilliams>
 - 181 <http://en.wikibooks.org/w/index.php?title=User:Daworm>
 - 182 <http://en.wikibooks.org/w/index.php?title=User:DbD>
 - 183 <http://en.wikibooks.org/w/index.php?title=User:Dbproguy>
 - 184 <http://en.wikibooks.org/w/index.php?title=User:Dbreece>
 - 185 <http://en.wikibooks.org/w/index.php?title=User:DeltaSpeeds>
 - 186 <http://en.wikibooks.org/w/index.php?title=User:Demoathon>
 - 187 http://en.wikibooks.org/w/index.php?title=User:Dennis_G._Jerz
 - 188 http://en.wikibooks.org/w/index.php?title=User:Der_K%C3%BCnstler
 - 189 <http://en.wikibooks.org/w/index.php?title=User:Derbeth>
 - 190 <http://en.wikibooks.org/w/index.php?title=User:Dessydes>
 - 191 <http://en.wikibooks.org/w/index.php?title=User:Diblidabliduu>
 - 192 <http://en.wikibooks.org/w/index.php?title=User:Digitalartist>
 - 193 <http://en.wikibooks.org/w/index.php?title=User:Digitalcircuit36939>
 - 194 <http://en.wikibooks.org/w/index.php?title=User:Digivation>
 - 195 http://en.wikibooks.org/w/index.php?title=User:Dimzy_Doodle
 - 196 <http://en.wikibooks.org/w/index.php?title=User:Dipingo>

3	Dirk Hünninger ¹⁹⁷
1	Dirk gently ¹⁹⁸
2	Ditw0101 ¹⁹⁹
1	Dkarpp ²⁰⁰
1	Dlr ²⁰¹
1	Dmansouth ²⁰²
2	Dmwick ²⁰³
5	Dobz116 ²⁰⁴
3	Dogwynn ²⁰⁵
1	Doj ^{o206}
4	Dolsson5 ²⁰⁷
1	Domthedude001 ²⁰⁸
9	Donlemay ²⁰⁹
8	Dooglus ²¹⁰
2	Dougc ²¹¹
2	Douglasfeb15 ²¹²
2	DrDixie ²¹³
3	Dragonkingme ²¹⁴
5	Drblitzkrieg ²¹⁵
2	Dreas ²¹⁶
5	Dspaun ²¹⁷
12	DuLithgow ²¹⁸
1	Dual Shock 2 ps2 controller ²¹⁹
3	Dudekahedron ²²⁰
2	EN-Eagle 101 ²²¹

197	http://en.wikibooks.org/w/index.php?title=User:Dirk_H%C3%BCnniger
198	http://en.wikibooks.org/w/index.php?title=User:Dirk_gently
199	http://en.wikibooks.org/w/index.php?title=User:Ditw0101
200	http://en.wikibooks.org/w/index.php?title=User:Dkarpp
201	http://en.wikibooks.org/w/index.php?title=User:Dlr
202	http://en.wikibooks.org/w/index.php?title=User:Dmansouth
203	http://en.wikibooks.org/w/index.php?title=User:Dmwick
204	http://en.wikibooks.org/w/index.php?title=User:Dobz116
205	http://en.wikibooks.org/w/index.php?title=User:Dogwynn
206	http://en.wikibooks.org/w/index.php?title=User:Doj%C2%B0
207	http://en.wikibooks.org/w/index.php?title=User:Dolsson5
208	http://en.wikibooks.org/w/index.php?title=User:Domthedude001
209	http://en.wikibooks.org/w/index.php?title=User:Donlemay
210	http://en.wikibooks.org/w/index.php?title=User:Dooglus
211	http://en.wikibooks.org/w/index.php?title=User:Dougc
212	http://en.wikibooks.org/w/index.php?title=User:Douglasfeb15
213	http://en.wikibooks.org/w/index.php?title=User:DrDixie
214	http://en.wikibooks.org/w/index.php?title=User:Dragonkingme
215	http://en.wikibooks.org/w/index.php?title=User:Drblitzkrieg
216	http://en.wikibooks.org/w/index.php?title=User:Dreas
217	http://en.wikibooks.org/w/index.php?title=User:Dspaun
218	http://en.wikibooks.org/w/index.php?title=User:DuLithgow
219	http://en.wikibooks.org/w/index.php?title=User:Dual_Shock_2_ps2_controller
220	http://en.wikibooks.org/w/index.php?title=User:Dudekahedron
221	http://en.wikibooks.org/w/index.php?title=User:EN-Eagle_101

6	EatMyShortz ²²²
6	Eb264 ²²³
11	Ebswift ²²⁴
1	Edmetric ²²⁵
3	Edr ²²⁶
1	Efren ²²⁷
1	Ekricyote ²²⁸
4	El imp ²²⁹
1	Electrice ²³⁰
16	Electro ²³¹
4	Electus ²³²
12	Embri ²³³
1	Encesil ²³⁴
2	Enochsdream ²³⁵
1	Enselic ²³⁶
2	Entera voss ²³⁷
2	Epicor ²³⁸
14	ErikBongers ²³⁹
2	Eruantalon ²⁴⁰
2	Esben ²⁴¹
147	EvanCarroll ²⁴²
1	Exo ²⁴³
3	Eyeonus ²⁴⁴
2	Falconx1096 ²⁴⁵
1	Falke ²⁴⁶

222 <http://en.wikibooks.org/w/index.php?title=User:EatMyShortz>
223 <http://en.wikibooks.org/w/index.php?title=User:Eb264>
224 <http://en.wikibooks.org/w/index.php?title=User:Ebswift>
225 <http://en.wikibooks.org/w/index.php?title=User:Edmetric>
226 <http://en.wikibooks.org/w/index.php?title=User:Edr>
227 <http://en.wikibooks.org/w/index.php?title=User:Efren>
228 <http://en.wikibooks.org/w/index.php?title=User:Ekricyote>
229 http://en.wikibooks.org/w/index.php?title=User:El_imp
230 <http://en.wikibooks.org/w/index.php?title=User:Electrice>
231 <http://en.wikibooks.org/w/index.php?title=User:Electro>
232 <http://en.wikibooks.org/w/index.php?title=User:Electus>
233 <http://en.wikibooks.org/w/index.php?title=User:Embri>
234 <http://en.wikibooks.org/w/index.php?title=User:Encesil>
235 <http://en.wikibooks.org/w/index.php?title=User:Enochsdream>
236 <http://en.wikibooks.org/w/index.php?title=User:Enselic>
237 http://en.wikibooks.org/w/index.php?title=User:Entera_voss
238 <http://en.wikibooks.org/w/index.php?title=User:Epicor>
239 <http://en.wikibooks.org/w/index.php?title=User:ErikBongers>
240 <http://en.wikibooks.org/w/index.php?title=User:Eruantalon>
241 <http://en.wikibooks.org/w/index.php?title=User:Esben>
242 <http://en.wikibooks.org/w/index.php?title=User:EvanCarroll>
243 <http://en.wikibooks.org/w/index.php?title=User:Exo>
244 <http://en.wikibooks.org/w/index.php?title=User:Eyeonus>
245 <http://en.wikibooks.org/w/index.php?title=User:Falconx1096>
246 <http://en.wikibooks.org/w/index.php?title=User:Falke>

8	Falstaff ²⁴⁷
5	Fangiotophia ²⁴⁸
1	Faux ²⁴⁹
9	Fede.Campana ²⁵⁰
1	Fedexdoom ²⁵¹
1	Feithless ²⁵²
3	Feureau ²⁵³
5	Fezz ²⁵⁴
4	Fidoda ²⁵⁵
2	Filaminae ²⁵⁶
1	Firecentaur ²⁵⁷
1	FishFace ²⁵⁸
1	Fishpi ²⁵⁹
1	Flavoice ²⁶⁰
1	Frank H ²⁶¹
5	FranklyMyDear... ²⁶²
2	Freakazo v2.1 ²⁶³
2	Freis ²⁶⁴
1	FriedChiliDog1 ²⁶⁵
2	Ftiercel ²⁶⁶
9	Funky weasel ²⁶⁷
6	Furrykef ²⁶⁸
6	Gaandolf ²⁶⁹
54	Gabio ²⁷⁰
5	Gairlochan ²⁷¹

247	http://en.wikibooks.org/w/index.php?title=User:Falstaff
248	http://en.wikibooks.org/w/index.php?title=User:Fangiotophia
249	http://en.wikibooks.org/w/index.php?title=User:Faux
250	http://en.wikibooks.org/w/index.php?title=User:Fede.Campana
251	http://en.wikibooks.org/w/index.php?title=User:Fedexdoom
252	http://en.wikibooks.org/w/index.php?title=User:Feithless
253	http://en.wikibooks.org/w/index.php?title=User:Feureau
254	http://en.wikibooks.org/w/index.php?title=User:Fezz
255	http://en.wikibooks.org/w/index.php?title=User:Fidoda
256	http://en.wikibooks.org/w/index.php?title=User:Filaminae
257	http://en.wikibooks.org/w/index.php?title=User:Firecentaur
258	http://en.wikibooks.org/w/index.php?title=User:FishFace
259	http://en.wikibooks.org/w/index.php?title=User:Fishpi
260	http://en.wikibooks.org/w/index.php?title=User:Flavoice
261	http://en.wikibooks.org/w/index.php?title=User:Frank_H
262	http://en.wikibooks.org/w/index.php?title=User:FranklyMyDear...
263	http://en.wikibooks.org/w/index.php?title=User:Freakazo_v2.1
264	http://en.wikibooks.org/w/index.php?title=User:Freis
265	http://en.wikibooks.org/w/index.php?title=User:FriedChiliDog1
266	http://en.wikibooks.org/w/index.php?title=User:Ftiercel
267	http://en.wikibooks.org/w/index.php?title=User:Funky_weasel
268	http://en.wikibooks.org/w/index.php?title=User:Furrykef
269	http://en.wikibooks.org/w/index.php?title=User:Gaandolf
270	http://en.wikibooks.org/w/index.php?title=User:Gabio
271	http://en.wikibooks.org/w/index.php?title=User:Gairlochan

3	Garmon Sutrix ²⁷²
1	Gavinmusic ²⁷³
9	GeForce3 ²⁷⁴
1	Geekboy2000 ²⁷⁵
1	Geigerho ²⁷⁶
2	GeniXPro ²⁷⁷
10	Geocachernemesis ²⁷⁸
2	Geoff Plourde ²⁷⁹
1	GeordieMcBain ²⁸⁰
6	Geri ²⁸¹
1	Ggolem1 ²⁸²
1	Ghiraddje ²⁸³
1	Gijs.peek ²⁸⁴
7	Gilgamesh007 ²⁸⁵
1	Girdi ²⁸⁶
2	Glome ²⁸⁷
2	Goatofdeath ²⁸⁸
1	Goeland86 ²⁸⁹
1	Goncalopp ²⁹⁰
4	GreenBlobWiki ²⁹¹
11	Greenbreen ²⁹²
6	Greenlid ²⁹³
32	Greensweater ²⁹⁴
12	Greggreg ²⁹⁵
1	Grubham ²⁹⁶

272	http://en.wikibooks.org/w/index.php?title=User:Garmon_Sutrix
273	http://en.wikibooks.org/w/index.php?title=User:Gavinmusic
274	http://en.wikibooks.org/w/index.php?title=User:GeForce3
275	http://en.wikibooks.org/w/index.php?title=User:Geekboy2000
276	http://en.wikibooks.org/w/index.php?title=User:Geigerho
277	http://en.wikibooks.org/w/index.php?title=User:GeniXPro
278	http://en.wikibooks.org/w/index.php?title=User:Geocachernemesis
279	http://en.wikibooks.org/w/index.php?title=User:Geoff_Plourde
280	http://en.wikibooks.org/w/index.php?title=User:GeordieMcBain
281	http://en.wikibooks.org/w/index.php?title=User:Geri
282	http://en.wikibooks.org/w/index.php?title=User:Ggolem1
283	http://en.wikibooks.org/w/index.php?title=User:Ghiraddje
284	http://en.wikibooks.org/w/index.php?title=User:Gijs.peek
285	http://en.wikibooks.org/w/index.php?title=User:Gilgamesh007
286	http://en.wikibooks.org/w/index.php?title=User:Girdi
287	http://en.wikibooks.org/w/index.php?title=User:Glome
288	http://en.wikibooks.org/w/index.php?title=User:Goatofdeath
289	http://en.wikibooks.org/w/index.php?title=User:Goeland86
290	http://en.wikibooks.org/w/index.php?title=User:Goncalopp
291	http://en.wikibooks.org/w/index.php?title=User:GreenBlobWiki
292	http://en.wikibooks.org/w/index.php?title=User:Greenbreen
293	http://en.wikibooks.org/w/index.php?title=User:Greenlid
294	http://en.wikibooks.org/w/index.php?title=User:Greensweater
295	http://en.wikibooks.org/w/index.php?title=User:Greggreg
296	http://en.wikibooks.org/w/index.php?title=User:Grubham

- 2 Gumanoed²⁹⁷
- 2 Gumba gumba²⁹⁸
- 1 GustavTheMushroom²⁹⁹
- 2 HHBones³⁰⁰
- 1 HWV258³⁰¹
- 1 Habstinat³⁰²
- 1 Hadphild³⁰³
- 7 Hagindaz³⁰⁴
- 16 Halley³⁰⁵
- 1 Hammurderer³⁰⁶
- 1 Happy2pester³⁰⁷
- 3 Harkin³⁰⁸
- 4 Hash³⁰⁹
- 1 Hbrown101³¹⁰
- 2 Heinrich5991³¹¹
- 1 Helderpc³¹²
- 1 Hendric³¹³
- 8 Henrybissonnette³¹⁴
- 17 Herbythyme³¹⁵
- 2 Herraotic³¹⁶
- 3 HethrirBot³¹⁷
- 2 Hippopotamuses³¹⁸
- 1 Hobo Joe³¹⁹
- 1 Hom sepanta³²⁰
- 2 Hoo man³²¹

-
- 297 <http://en.wikibooks.org/w/index.php?title=User:Gumanoed>
 - 298 http://en.wikibooks.org/w/index.php?title=User:Gumba_gumba
 - 299 <http://en.wikibooks.org/w/index.php?title=User:GustavTheMushroom>
 - 300 <http://en.wikibooks.org/w/index.php?title=User:HHBones>
 - 301 <http://en.wikibooks.org/w/index.php?title=User:HWV258>
 - 302 <http://en.wikibooks.org/w/index.php?title=User:Habstinat>
 - 303 <http://en.wikibooks.org/w/index.php?title=User:Hadphild>
 - 304 <http://en.wikibooks.org/w/index.php?title=User:Hagindaz>
 - 305 <http://en.wikibooks.org/w/index.php?title=User:Halley>
 - 306 <http://en.wikibooks.org/w/index.php?title=User:Hammurderer>
 - 307 <http://en.wikibooks.org/w/index.php?title=User:Happy2pester>
 - 308 <http://en.wikibooks.org/w/index.php?title=User:Harkin>
 - 309 <http://en.wikibooks.org/w/index.php?title=User:Hash>
 - 310 <http://en.wikibooks.org/w/index.php?title=User:Hbrown101>
 - 311 <http://en.wikibooks.org/w/index.php?title=User:Heinrich5991>
 - 312 <http://en.wikibooks.org/w/index.php?title=User:Helderpc>
 - 313 <http://en.wikibooks.org/w/index.php?title=User:Hendric>
 - 314 <http://en.wikibooks.org/w/index.php?title=User:Henrybissonnette>
 - 315 <http://en.wikibooks.org/w/index.php?title=User:Herbythyme>
 - 316 <http://en.wikibooks.org/w/index.php?title=User:Herraotic>
 - 317 <http://en.wikibooks.org/w/index.php?title=User:HethrirBot>
 - 318 <http://en.wikibooks.org/w/index.php?title=User:Hippopotamuses>
 - 319 http://en.wikibooks.org/w/index.php?title=User:Hobo_Joe
 - 320 http://en.wikibooks.org/w/index.php?title=User:Hom_sepanta
 - 321 http://en.wikibooks.org/w/index.php?title=User:Hoo_man

6	Hrakaroo ³²²
1	Hxcan ³²³
1	Hydriz ³²⁴
5	IamInnocent ³²⁵
3	Iamslite ³²⁶
5	Iamunknown ³²⁷
1	Ibid49 ³²⁸
7	Icky-wiki ³²⁹
5	Ideasman42 ³³⁰
1	Iguimaraesf ³³¹
3	Illusionist ³³²
8	Iluvblender ³³³
2	Imaginationac ³³⁴
1	India103 ³³⁵
1	Inductiveload ³³⁶
1	Inityx ³³⁷
4	Inquisitor Kaoz ³³⁸
1	Intuitas ³³⁹
1	Inventor ³⁴⁰
10	IqAndreas ³⁴¹
13	Irakrakow ³⁴²
1	Ironicaby ³⁴³
1	Itineranttrader ³⁴⁴
1	Itismike ³⁴⁵
5	Ivucica ³⁴⁶

322 <http://en.wikibooks.org/w/index.php?title=User:Hrakaroo>
323 <http://en.wikibooks.org/w/index.php?title=User:Hxcan>
324 <http://en.wikibooks.org/w/index.php?title=User:Hydriz>
325 <http://en.wikibooks.org/w/index.php?title=User:IamInnocent>
326 <http://en.wikibooks.org/w/index.php?title=User:Iamslite>
327 <http://en.wikibooks.org/w/index.php?title=User:Iamunknown>
328 <http://en.wikibooks.org/w/index.php?title=User:Ibid49>
329 <http://en.wikibooks.org/w/index.php?title=User:Icky-wiki>
330 <http://en.wikibooks.org/w/index.php?title=User:Ideasman42>
331 <http://en.wikibooks.org/w/index.php?title=User:Iguimaraesf>
332 <http://en.wikibooks.org/w/index.php?title=User:Illusionist>
333 <http://en.wikibooks.org/w/index.php?title=User:Iluvblender>
334 <http://en.wikibooks.org/w/index.php?title=User:Imaginationac>
335 <http://en.wikibooks.org/w/index.php?title=User:India103>
336 <http://en.wikibooks.org/w/index.php?title=User:Inductiveload>
337 <http://en.wikibooks.org/w/index.php?title=User:Inityx>
338 http://en.wikibooks.org/w/index.php?title=User:Inquisitor_Kaoz
339 <http://en.wikibooks.org/w/index.php?title=User:Intuitas>
340 <http://en.wikibooks.org/w/index.php?title=User:Inventor>
341 <http://en.wikibooks.org/w/index.php?title=User:IqAndreas>
342 <http://en.wikibooks.org/w/index.php?title=User:Irakrakow>
343 <http://en.wikibooks.org/w/index.php?title=User:Ironicaby>
344 <http://en.wikibooks.org/w/index.php?title=User:Itineranttrader>
345 <http://en.wikibooks.org/w/index.php?title=User:Itismike>
346 <http://en.wikibooks.org/w/index.php?title=User:Ivucica>

- 2 J36miles³⁴⁷
- 1 JCrawford³⁴⁸
- 1 JHeathe³⁴⁹
- 4 JRthro³⁵⁰
- 1 JacKDuRdEn³⁵¹
- 2 JaccoG³⁵²
- 27 Jackechan³⁵³
- 1 Jackred³⁵⁴
- 1 Jake Scordato³⁵⁵
- 3 Jake.lewis4³⁵⁶
- 2 James.Denholm³⁵⁷
- 1 JamesCrook³⁵⁸
- 65 JamesNZ³⁵⁹
- 7 Jamessnell³⁶⁰
- 2 Jamin3D³⁶¹
- 1 Javac³⁶²
- 9 Jawboot³⁶³
- 1 Jbalint³⁶⁴
- 5 Jclee³⁶⁵
- 2 Jensenns³⁶⁶
- 5 Jeremy³⁶⁷
- 1 Jeromerobert³⁶⁸
- 1 Jessemerriman³⁶⁹
- 1 Jfireball66³⁷⁰
- 1 Jfmantis³⁷¹

347 <http://en.wikibooks.org/w/index.php?title=User:J36miles>

348 <http://en.wikibooks.org/w/index.php?title=User:JCrawford>

349 <http://en.wikibooks.org/w/index.php?title=User:JHeathe>

350 <http://en.wikibooks.org/w/index.php?title=User:JRthro>

351 <http://en.wikibooks.org/w/index.php?title=User:JacKDuRdEn>

352 <http://en.wikibooks.org/w/index.php?title=User:JaccoG>

353 <http://en.wikibooks.org/w/index.php?title=User:Jackechan>

354 <http://en.wikibooks.org/w/index.php?title=User:Jackred>

355 http://en.wikibooks.org/w/index.php?title=User:Jake_Scordato

356 <http://en.wikibooks.org/w/index.php?title=User:Jake.lewis4>

357 <http://en.wikibooks.org/w/index.php?title=User:James.Denholm>

358 <http://en.wikibooks.org/w/index.php?title=User:JamesCrook>

359 <http://en.wikibooks.org/w/index.php?title=User:JamesNZ>

360 <http://en.wikibooks.org/w/index.php?title=User:Jamessnell>

361 <http://en.wikibooks.org/w/index.php?title=User:Jamin3D>

362 <http://en.wikibooks.org/w/index.php?title=User:Javac>

363 <http://en.wikibooks.org/w/index.php?title=User:Jawboot>

364 <http://en.wikibooks.org/w/index.php?title=User:Jbalint>

365 <http://en.wikibooks.org/w/index.php?title=User:Jclee>

366 <http://en.wikibooks.org/w/index.php?title=User:Jensenns>

367 <http://en.wikibooks.org/w/index.php?title=User:Jeremy>

368 <http://en.wikibooks.org/w/index.php?title=User:Jeromerobert>

369 <http://en.wikibooks.org/w/index.php?title=User:Jessemerriman>

370 <http://en.wikibooks.org/w/index.php?title=User:Jfireball66>

371 <http://en.wikibooks.org/w/index.php?title=User:Jfmantis>

- 73 Jguk³⁷²
- 1 JiBril³⁷³
- 8 JimRester³⁷⁴
- 1 Jleedev³⁷⁵
- 5 Jmartens³⁷⁶
- 6 Joda³⁷⁷
- 4 Joe 042293³⁷⁸
- 1 Joeblends³⁷⁹
- 5 JoelGodin³⁸⁰
- 1 Jogai³⁸¹
- 6 John Nagle³⁸²
- 2 John.delannoy³⁸³
- 1 Johndb³⁸⁴
- 3 Johnno³⁸⁵
- 1 Johnstarbird³⁸⁶
- 2 Jollyroger³⁸⁷
- 47 Jomegat³⁸⁸
- 2 Joncomp12³⁸⁹
- 1 Jonno333³⁹⁰
- 19 Jonon³⁹¹
- 1 Jorge Morais³⁹²
- 4 Jose da vinci³⁹³
- 3 Josellis³⁹⁴
- 15 Joystick Game³⁹⁵
- 1 Jsrowndog³⁹⁶

-
- 372 <http://en.wikibooks.org/w/index.php?title=User:Jguk>
 - 373 <http://en.wikibooks.org/w/index.php?title=User:JiBril>
 - 374 <http://en.wikibooks.org/w/index.php?title=User:JimRester>
 - 375 <http://en.wikibooks.org/w/index.php?title=User:Jleedev>
 - 376 <http://en.wikibooks.org/w/index.php?title=User:Jmartens>
 - 377 <http://en.wikibooks.org/w/index.php?title=User:Joda>
 - 378 http://en.wikibooks.org/w/index.php?title=User:Joe_042293
 - 379 <http://en.wikibooks.org/w/index.php?title=User:Joeblends>
 - 380 <http://en.wikibooks.org/w/index.php?title=User:JoelGodin>
 - 381 <http://en.wikibooks.org/w/index.php?title=User:Jogai>
 - 382 http://en.wikibooks.org/w/index.php?title=User:John_Nagle
 - 383 <http://en.wikibooks.org/w/index.php?title=User:John.delannoy>
 - 384 <http://en.wikibooks.org/w/index.php?title=User:Johndb>
 - 385 <http://en.wikibooks.org/w/index.php?title=User:Johnno>
 - 386 <http://en.wikibooks.org/w/index.php?title=User:Johnstarbird>
 - 387 <http://en.wikibooks.org/w/index.php?title=User:Jollyroger>
 - 388 <http://en.wikibooks.org/w/index.php?title=User:Jomegat>
 - 389 <http://en.wikibooks.org/w/index.php?title=User:Joncomp12>
 - 390 <http://en.wikibooks.org/w/index.php?title=User:Jonno333>
 - 391 <http://en.wikibooks.org/w/index.php?title=User:Jonon>
 - 392 http://en.wikibooks.org/w/index.php?title=User:Jorge_Morais
 - 393 http://en.wikibooks.org/w/index.php?title=User:Jose_da_vinci
 - 394 <http://en.wikibooks.org/w/index.php?title=User:Josellis>
 - 395 http://en.wikibooks.org/w/index.php?title=User:Joystick_Game
 - 396 <http://en.wikibooks.org/w/index.php?title=User:Jsrowndog>

- 1 Jshadias³⁹⁷
- 1 Jtonti³⁹⁸
- 2 JuBlender³⁹⁹
- 2 Justin6300⁴⁰⁰
- 1 JustinTime55⁴⁰¹
- 3 Justlost⁴⁰²
- 1 Justthisguy⁴⁰³
- 1 Jyri1⁴⁰⁴
- 1 KaiRyu⁴⁰⁵
- 1 Kakurady⁴⁰⁶
- 4 Kayau⁴⁰⁷
- 3 Kdjerdon⁴⁰⁸
- 14 Keki1⁴⁰⁹
- 8 Kenkaku⁴¹⁰
- 2 KentTong⁴¹¹
- 5 Kerina yin⁴¹²
- 1 Kernigh⁴¹³
- 4 Kesten⁴¹⁴
- 5 Khan⁴¹⁵
- 4 Khono⁴¹⁶
- 1 Kieran⁴¹⁷
- 4 Kindeller⁴¹⁸
- 9 Kirona⁴¹⁹
- 6 Kisubo⁴²⁰
- 14 Klodge⁴²¹

397 <http://en.wikibooks.org/w/index.php?title=User:Jshadias>
398 <http://en.wikibooks.org/w/index.php?title=User:Jtonti>
399 <http://en.wikibooks.org/w/index.php?title=User:JuBlender>
400 <http://en.wikibooks.org/w/index.php?title=User:Justin6300>
401 <http://en.wikibooks.org/w/index.php?title=User:JustinTime55>
402 <http://en.wikibooks.org/w/index.php?title=User:Justlost>
403 <http://en.wikibooks.org/w/index.php?title=User:Justthisguy>
404 <http://en.wikibooks.org/w/index.php?title=User:Jyri1>
405 <http://en.wikibooks.org/w/index.php?title=User:KaiRyu>
406 <http://en.wikibooks.org/w/index.php?title=User:Kakurady>
407 <http://en.wikibooks.org/w/index.php?title=User:Kayau>
408 <http://en.wikibooks.org/w/index.php?title=User:Kdjerdon>
409 <http://en.wikibooks.org/w/index.php?title=User:Keki1>
410 <http://en.wikibooks.org/w/index.php?title=User:Kengkaku>
411 <http://en.wikibooks.org/w/index.php?title=User:KentTong>
412 http://en.wikibooks.org/w/index.php?title=User:Kerina_yin
413 <http://en.wikibooks.org/w/index.php?title=User:Kernigh>
414 <http://en.wikibooks.org/w/index.php?title=User:Kesten>
415 <http://en.wikibooks.org/w/index.php?title=User:Khan>
416 <http://en.wikibooks.org/w/index.php?title=User:Khono>
417 <http://en.wikibooks.org/w/index.php?title=User:Kieran>
418 <http://en.wikibooks.org/w/index.php?title=User:Kindeller>
419 <http://en.wikibooks.org/w/index.php?title=User:Kirona>
420 <http://en.wikibooks.org/w/index.php?title=User:Kisubo>
421 <http://en.wikibooks.org/w/index.php?title=User:Klodge>

2 Knight13⁴²²
1 Kok⁴²³
3 Koltonaugust⁴²⁴
1 Krand⁴²⁵
1 Krismania⁴²⁶
2 Ktbluear⁴²⁷
6 Kylstoman⁴²⁸
2 Laleena⁴²⁹
1 Lambda (Usurped)⁴³⁰
1 Lapinbleu⁴³¹
30 Lazy-lump⁴³²
348 Ldo⁴³³
9 Lebigot⁴³⁴
3 Lechtitseb⁴³⁵
23 Lee Carre⁴³⁶
12 LeftClicker⁴³⁷
1 Leinad13⁴³⁸
1 LethalReflex⁴³⁹
1 Lethargo⁴⁴⁰
1 Levis501⁴⁴¹
5 Lionelb⁴⁴²
12 Lirtferk⁴⁴³
47 LokiClock⁴⁴⁴
1 Lokimaros⁴⁴⁵
4 Longsword⁴⁴⁶

422 <http://en.wikibooks.org/w/index.php?title=User:Knight13>
423 <http://en.wikibooks.org/w/index.php?title=User:Kok>
424 <http://en.wikibooks.org/w/index.php?title=User:Koltonaugust>
425 <http://en.wikibooks.org/w/index.php?title=User:Krand>
426 <http://en.wikibooks.org/w/index.php?title=User:Krismania>
427 <http://en.wikibooks.org/w/index.php?title=User:Ktbluear>
428 <http://en.wikibooks.org/w/index.php?title=User:Kylstoman>
429 <http://en.wikibooks.org/w/index.php?title=User:Laleena>
430 http://en.wikibooks.org/w/index.php?title=User:Lambda_%28Usurped%29
431 <http://en.wikibooks.org/w/index.php?title=User:Lapinbleu>
432 <http://en.wikibooks.org/w/index.php?title=User:Lazy-lump>
433 <http://en.wikibooks.org/w/index.php?title=User:Ldo>
434 <http://en.wikibooks.org/w/index.php?title=User:Lebigot>
435 <http://en.wikibooks.org/w/index.php?title=User:Lechtitseb>
436 http://en.wikibooks.org/w/index.php?title=User:Lee_Carre
437 <http://en.wikibooks.org/w/index.php?title=User:LeftClicker>
438 <http://en.wikibooks.org/w/index.php?title=User:Leinad13>
439 <http://en.wikibooks.org/w/index.php?title=User:LethalReflex>
440 <http://en.wikibooks.org/w/index.php?title=User:Lethargo>
441 <http://en.wikibooks.org/w/index.php?title=User:Levis501>
442 <http://en.wikibooks.org/w/index.php?title=User:Lionelb>
443 <http://en.wikibooks.org/w/index.php?title=User:Lirtferk>
444 <http://en.wikibooks.org/w/index.php?title=User:LokiClock>
445 <http://en.wikibooks.org/w/index.php?title=User:Lokimaros>
446 <http://en.wikibooks.org/w/index.php?title=User:Longsword>

- 11 Looterguf⁴⁴⁷
- 1 Loplin⁴⁴⁸
- 6 LordOfer⁴⁴⁹
- 4 Lordelph⁴⁵⁰
- 1 Lordmyth⁴⁵¹
- 2 Lordscapes91⁴⁵²
- 1 Lorhc⁴⁵³
- 1 Lowbot0⁴⁵⁴
- 1 Lsmoura⁴⁵⁵
- 1 Luckofbuck⁴⁵⁶
- 1 Luke321321⁴⁵⁷
- 1 Lxxxv⁴⁵⁸
- 1 Lynx7725⁴⁵⁹
- 1 M.Nießen⁴⁶⁰
- 1 MC10⁴⁶¹
- 12 MSK61⁴⁶²
- 7 Ma5abre⁴⁶³
- 1 Mabdul⁴⁶⁴
- 2 MacPepe⁴⁶⁵
- 4 MacPhyle⁴⁶⁶
- 1 Mach1723⁴⁶⁷
- 13 Madcello⁴⁶⁸
- 1 Maebe⁴⁶⁹
- 1 Majko02⁴⁷⁰
- 81 Malaz⁴⁷¹

447 <http://en.wikibooks.org/w/index.php?title=User:Looterguf>

448 <http://en.wikibooks.org/w/index.php?title=User:Loplin>

449 <http://en.wikibooks.org/w/index.php?title=User:LordOfer>

450 <http://en.wikibooks.org/w/index.php?title=User:Lordelph>

451 <http://en.wikibooks.org/w/index.php?title=User:Lordmyth>

452 <http://en.wikibooks.org/w/index.php?title=User:Lordscapes91>

453 <http://en.wikibooks.org/w/index.php?title=User:Lorhc>

454 <http://en.wikibooks.org/w/index.php?title=User:Lowbot0>

455 <http://en.wikibooks.org/w/index.php?title=User:Lsmoura>

456 <http://en.wikibooks.org/w/index.php?title=User:Luckofbuck>

457 <http://en.wikibooks.org/w/index.php?title=User:Luke321321>

458 <http://en.wikibooks.org/w/index.php?title=User:Lxxxv>

459 <http://en.wikibooks.org/w/index.php?title=User:Lynx7725>

460 <http://en.wikibooks.org/w/index.php?title=User:M.Nie%C3%9Fen>

461 <http://en.wikibooks.org/w/index.php?title=User:MC10>

462 <http://en.wikibooks.org/w/index.php?title=User:MSK61>

463 <http://en.wikibooks.org/w/index.php?title=User:Ma5abre>

464 <http://en.wikibooks.org/w/index.php?title=User:Mabdul>

465 <http://en.wikibooks.org/w/index.php?title=User:MacPepe>

466 <http://en.wikibooks.org/w/index.php?title=User:MacPhyle>

467 <http://en.wikibooks.org/w/index.php?title=User:Mach1723>

468 <http://en.wikibooks.org/w/index.php?title=User:Madcello>

469 <http://en.wikibooks.org/w/index.php?title=User:Maebe>

470 <http://en.wikibooks.org/w/index.php?title=User:Majko02>

471 <http://en.wikibooks.org/w/index.php?title=User:Malaz>

2	Malvineous ⁴⁷²
1	Maniandram01 ⁴⁷³
1	Manx61 ⁴⁷⁴
1	Marco Guardigli ⁴⁷⁵
2	Marinalan ⁴⁷⁶
1	MarkLorenWilson ⁴⁷⁷
2	MarkM ⁴⁷⁸
3	Markharper80 ⁴⁷⁹
3	Marr Gilb ⁴⁸⁰
1	Marty one ⁴⁸¹
15	Mast3rlinkx ⁴⁸²
1	Mats Halldin ⁴⁸³
5	Matt Howard ⁴⁸⁴
1	Matt Pellegrini ⁴⁸⁵
1	Mattb112885 ⁴⁸⁶
1	Matthewr100 ⁴⁸⁷
1	Matthewrbowker ⁴⁸⁸
3	Mattiasl ⁴⁸⁹
1	Maveric149 ⁴⁹⁰
1	Mavery1 ⁴⁹¹
16	Mdd4696 ⁴⁹²
6	Mecalith ⁴⁹³
2	MechaBlue ⁴⁹⁴
3	Megaloman ⁴⁹⁵
1	Memiles ⁴⁹⁶

472 <http://en.wikibooks.org/w/index.php?title=User:Malvineous>
473 <http://en.wikibooks.org/w/index.php?title=User:Maniandram01>
474 <http://en.wikibooks.org/w/index.php?title=User:Manx61>
475 http://en.wikibooks.org/w/index.php?title=User:Marco_Guardigli
476 <http://en.wikibooks.org/w/index.php?title=User:Marinalan>
477 <http://en.wikibooks.org/w/index.php?title=User:MarkLorenWilson>
478 <http://en.wikibooks.org/w/index.php?title=User:MarkM>
479 <http://en.wikibooks.org/w/index.php?title=User:Markharper80>
480 http://en.wikibooks.org/w/index.php?title=User:Marr_Gilb
481 http://en.wikibooks.org/w/index.php?title=User:Marty_one
482 <http://en.wikibooks.org/w/index.php?title=User:Mast3rlinkx>
483 http://en.wikibooks.org/w/index.php?title=User:Mats_Halldin
484 http://en.wikibooks.org/w/index.php?title=User:Matt_Howard
485 http://en.wikibooks.org/w/index.php?title=User:Matt_Pellegrini
486 <http://en.wikibooks.org/w/index.php?title=User:Mattb112885>
487 <http://en.wikibooks.org/w/index.php?title=User:Matthewr100>
488 <http://en.wikibooks.org/w/index.php?title=User:Matthewrbowker>
489 <http://en.wikibooks.org/w/index.php?title=User:Mattiasl>
490 <http://en.wikibooks.org/w/index.php?title=User:Maveric149>
491 <http://en.wikibooks.org/w/index.php?title=User:Mavery1>
492 <http://en.wikibooks.org/w/index.php?title=User:Mdd4696>
493 <http://en.wikibooks.org/w/index.php?title=User:Mecalith>
494 <http://en.wikibooks.org/w/index.php?title=User:MechaBlue>
495 <http://en.wikibooks.org/w/index.php?title=User:Megaloman>
496 <http://en.wikibooks.org/w/index.php?title=User:Memiles>

- 9 Mercator⁴⁹⁷
- 6 Merrillee⁴⁹⁸
- 1 Metahop⁴⁹⁹
- 1 Mfoxdogg⁵⁰⁰
- 2 Mgv⁵⁰¹
- 4 Mhoram⁵⁰²
- 1 Michael Coupland⁵⁰³
- 1 MichaelFrey⁵⁰⁴
- 2 Michaelnelson⁵⁰⁵
- 17 Michaelsobers⁵⁰⁶
- 1 Michaelx153⁵⁰⁷
- 1 Mico⁵⁰⁸
- 2 MidnightLightning⁵⁰⁹
- 3 Miga⁵¹⁰
- 1 Miika22⁵¹¹
- 1 Mikahl⁵¹²
- 28 Mike.lifeguard⁵¹³
- 3 Mikeg8r⁵¹⁴
- 1 Miken700⁵¹⁵
- 1 Mikenube⁵¹⁶
- 2 Mikerancourt⁵¹⁷
- 4 Mikespedia⁵¹⁸
- 1 Milahu⁵¹⁹
- 7 Milky⁵²⁰
- 7 Mimooh⁵²¹

497 <http://en.wikibooks.org/w/index.php?title=User:Mercator>
 498 <http://en.wikibooks.org/w/index.php?title=User:Merrillee>
 499 <http://en.wikibooks.org/w/index.php?title=User:Metahop>
 500 <http://en.wikibooks.org/w/index.php?title=User:Mfoxdogg>
 501 <http://en.wikibooks.org/w/index.php?title=User:Mgv>
 502 <http://en.wikibooks.org/w/index.php?title=User:Mhoram>
 503 http://en.wikibooks.org/w/index.php?title=User:Michael_Coupland
 504 <http://en.wikibooks.org/w/index.php?title=User:MichaelFrey>
 505 <http://en.wikibooks.org/w/index.php?title=User:Michaelnelson>
 506 <http://en.wikibooks.org/w/index.php?title=User:Michaelsobers>
 507 <http://en.wikibooks.org/w/index.php?title=User:Michaelx153>
 508 <http://en.wikibooks.org/w/index.php?title=User:Mico>
 509 <http://en.wikibooks.org/w/index.php?title=User:MidnightLightning>
 510 <http://en.wikibooks.org/w/index.php?title=User:Miga>
 511 <http://en.wikibooks.org/w/index.php?title=User:Miika22>
 512 <http://en.wikibooks.org/w/index.php?title=User:Mikahl>
 513 <http://en.wikibooks.org/w/index.php?title=User:Mike.lifeguard>
 514 <http://en.wikibooks.org/w/index.php?title=User:Mikeg8r>
 515 <http://en.wikibooks.org/w/index.php?title=User:Miken700>
 516 <http://en.wikibooks.org/w/index.php?title=User:Mikenube>
 517 <http://en.wikibooks.org/w/index.php?title=User:Mikerancourt>
 518 <http://en.wikibooks.org/w/index.php?title=User:Mikespedia>
 519 <http://en.wikibooks.org/w/index.php?title=User:Milahu>
 520 <http://en.wikibooks.org/w/index.php?title=User:Milky>
 521 <http://en.wikibooks.org/w/index.php?title=User:Mimooh>

- 1 Mindbulletmatrix⁵²²
- 4 Miyazaki⁵²³
- 9 Mochimo⁵²⁴
- 1 Mohax⁵²⁵
- 50 Moohasha⁵²⁶
- 9 Moreejt⁵²⁷
- 13 Morerunes⁵²⁸
- 16 Morrolan⁵²⁹
- 1 Mortennobel⁵³⁰
- 1 Mosquito⁵³¹
- 1 Mouse among men⁵³²
- 1 Mouse is back⁵³³
- 2 Mouzi⁵³⁴
- 1 Mr Me⁵³⁵
- 3 Mr Speaker⁵³⁶
- 1 MrThreepwood⁵³⁷
- 11 Mrfelis⁵³⁸
- 2 Mrmagee2759⁵³⁹
- 1 Mrwilley⁵⁴⁰
- 16 Mstram⁵⁴¹
- 1 Msvbdev⁵⁴²
- 13 Mtjlt⁵⁴³
- 2 Mudpuddle⁵⁴⁴
- 3 Muimi⁵⁴⁵
- 1 Mxn⁵⁴⁶

-
- 522 <http://en.wikibooks.org/w/index.php?title=User:Mindbulletmatrix>
 - 523 <http://en.wikibooks.org/w/index.php?title=User:Miyazaki>
 - 524 <http://en.wikibooks.org/w/index.php?title=User:Mochimo>
 - 525 <http://en.wikibooks.org/w/index.php?title=User:Mohax>
 - 526 <http://en.wikibooks.org/w/index.php?title=User:Moohasha>
 - 527 <http://en.wikibooks.org/w/index.php?title=User:Moreejt>
 - 528 <http://en.wikibooks.org/w/index.php?title=User:Morerunes>
 - 529 <http://en.wikibooks.org/w/index.php?title=User:Morrolan>
 - 530 <http://en.wikibooks.org/w/index.php?title=User:Mortennobel>
 - 531 <http://en.wikibooks.org/w/index.php?title=User:Mosquito>
 - 532 http://en.wikibooks.org/w/index.php?title=User:Mouse_among_men
 - 533 http://en.wikibooks.org/w/index.php?title=User:Mouse_is_back
 - 534 <http://en.wikibooks.org/w/index.php?title=User:Mouzi>
 - 535 http://en.wikibooks.org/w/index.php?title=User:Mr_Me
 - 536 http://en.wikibooks.org/w/index.php?title=User:Mr_Speaker
 - 537 <http://en.wikibooks.org/w/index.php?title=User:MrThreepwood>
 - 538 <http://en.wikibooks.org/w/index.php?title=User:Mrfelis>
 - 539 <http://en.wikibooks.org/w/index.php?title=User:Mrmagee2759>
 - 540 <http://en.wikibooks.org/w/index.php?title=User:Mrwilley>
 - 541 <http://en.wikibooks.org/w/index.php?title=User:Mstram>
 - 542 <http://en.wikibooks.org/w/index.php?title=User:Msvbdev>
 - 543 <http://en.wikibooks.org/w/index.php?title=User:Mtjlt>
 - 544 <http://en.wikibooks.org/w/index.php?title=User:Mudpuddle>
 - 545 <http://en.wikibooks.org/w/index.php?title=User:Muimi>
 - 546 <http://en.wikibooks.org/w/index.php?title=User:Mxn>

- 1 Mystictim⁵⁴⁷
- 104 N comme Nul⁵⁴⁸
- 1 N00b Henry⁵⁴⁹
- 2 NODJKQR⁵⁵⁰
- 1 Nanki⁵⁵¹
- 14 Nanobug⁵⁵²
- 1 Nanodeath⁵⁵³
- 2 NathanStocks⁵⁵⁴
- 1 Nczempin⁵⁵⁵
- 2 Neilwightman⁵⁵⁶
- 2 Neko18⁵⁵⁷
- 1 Neoptolemus⁵⁵⁸
- 3 Nerdrabidmonkey⁵⁵⁹
- 1 Nessup⁵⁶⁰
- 3 Neurotic⁵⁶¹
- 2 Neverbirth⁵⁶²
- 3 Newbie⁵⁶³
- 2 Newbienmiss⁵⁶⁴
- 1 Nick chang207⁵⁶⁵
- 12 Nick.anderegg⁵⁶⁶
- 1 Nicoladie⁵⁶⁷
- 1 Nielmo⁵⁶⁸
- 1 Niks1024⁵⁶⁹
- 1 NinjaSaru⁵⁷⁰
- 1 NitishP⁵⁷¹

547 <http://en.wikibooks.org/w/index.php?title=User:Mystictim>
548 http://en.wikibooks.org/w/index.php?title=User:N_comme_Nul
549 http://en.wikibooks.org/w/index.php?title=User:N00b_Henry
550 <http://en.wikibooks.org/w/index.php?title=User:NODJKQR>
551 <http://en.wikibooks.org/w/index.php?title=User:Nanki>
552 <http://en.wikibooks.org/w/index.php?title=User:Nanobug>
553 <http://en.wikibooks.org/w/index.php?title=User:Nanodeath>
554 <http://en.wikibooks.org/w/index.php?title=User:NathanStocks>
555 <http://en.wikibooks.org/w/index.php?title=User:Nczempin>
556 <http://en.wikibooks.org/w/index.php?title=User:Neilwightman>
557 <http://en.wikibooks.org/w/index.php?title=User:Neko18>
558 <http://en.wikibooks.org/w/index.php?title=User:Neoptolemus>
559 <http://en.wikibooks.org/w/index.php?title=User:Nerdrabidmonkey>
560 <http://en.wikibooks.org/w/index.php?title=User:Nessup>
561 <http://en.wikibooks.org/w/index.php?title=User:Neurotic>
562 <http://en.wikibooks.org/w/index.php?title=User:Neverbirth>
563 <http://en.wikibooks.org/w/index.php?title=User:Newbie>
564 <http://en.wikibooks.org/w/index.php?title=User:Newbienmiss>
565 http://en.wikibooks.org/w/index.php?title=User:Nick_chang207%60
566 <http://en.wikibooks.org/w/index.php?title=User:Nick.anderegg>
567 <http://en.wikibooks.org/w/index.php?title=User:Nicoladie>
568 <http://en.wikibooks.org/w/index.php?title=User:Nielmo>
569 <http://en.wikibooks.org/w/index.php?title=User:Niks1024>
570 <http://en.wikibooks.org/w/index.php?title=User:NinjaSaru>
571 <http://en.wikibooks.org/w/index.php?title=User:NitishP>

1	Nk.warren	77 ⁵⁷²
17	Nkansahreford	573
1	Noob_sks_dk	574
4	Norleaf	575
2	Nowic	576
1	NuclearWarfare	577
7	Nuclearpidgeon	578
47	Null_Point	579
3	NullPtr	580
3	Oasiac	581
2	Obiesel	582
11	Octava	583
5	Odalcet	584
5	Odie5533	585
1	Ok	586
1	Omnigeek6	587
2	Opa_God	588
12	Oracleoftruth	589
12	Orbisonitrum	590
2	Ordigdug	591
10	Orfo	592
3	Orindlincoln	593
1	Oshaylinux	594
1	Osxrules	595
5	Othmanskn	596

572 <http://en.wikibooks.org/w/index.php?title=User:Nk.warren77>
573 <http://en.wikibooks.org/w/index.php?title=User:Nkansahreford>
574 http://en.wikibooks.org/w/index.php?title=User:Noob_sks_dk
575 <http://en.wikibooks.org/w/index.php?title=User:Norleaf>
576 <http://en.wikibooks.org/w/index.php?title=User:Nowic>
577 <http://en.wikibooks.org/w/index.php?title=User:NuclearWarfare>
578 <http://en.wikibooks.org/w/index.php?title=User:Nuclearpidgeon>
579 http://en.wikibooks.org/w/index.php?title=User:Null_Point
580 <http://en.wikibooks.org/w/index.php?title=User:NullPtr>
581 <http://en.wikibooks.org/w/index.php?title=User:Oasiac>
582 <http://en.wikibooks.org/w/index.php?title=User:Obiesel>
583 <http://en.wikibooks.org/w/index.php?title=User:Octava>
584 <http://en.wikibooks.org/w/index.php?title=User:Odalcet>
585 <http://en.wikibooks.org/w/index.php?title=User:Odie5533>
586 <http://en.wikibooks.org/w/index.php?title=User:Ok>
587 <http://en.wikibooks.org/w/index.php?title=User:Omnigeek6>
588 http://en.wikibooks.org/w/index.php?title=User:Opa_God
589 <http://en.wikibooks.org/w/index.php?title=User:Oracleoftruth>
590 <http://en.wikibooks.org/w/index.php?title=User:Orbisonitrum>
591 <http://en.wikibooks.org/w/index.php?title=User:Ordigdug>
592 <http://en.wikibooks.org/w/index.php?title=User:Orfo>
593 <http://en.wikibooks.org/w/index.php?title=User:Orindlincoln>
594 <http://en.wikibooks.org/w/index.php?title=User:Oshaylinux>
595 <http://en.wikibooks.org/w/index.php?title=User:Osxrules>
596 <http://en.wikibooks.org/w/index.php?title=User:Othmanskn>

- 18 Othmansk2⁵⁹⁷
- 4 Outbacksam34⁵⁹⁸
- 1 Ownthor⁵⁹⁹
- 1 Ozgrin⁶⁰⁰
- 1 PKHG⁶⁰¹
- 4 POpZ⁶⁰²
- 3 Pamtango⁶⁰³
- 18 Panic2k4⁶⁰⁴
- 2 Paperclip⁶⁰⁵
- 15 ParallaxTZ⁶⁰⁶
- 2 PartierSP⁶⁰⁷
- 4 Patrolboat⁶⁰⁸
- 10 PaulJC⁶⁰⁹
- 1 Pavroo⁶¹⁰
- 611 Pearts⁶¹¹
- 11 Pedro Fonini⁶¹²
- 2 Pengo⁶¹³
- 1 Petariot⁶¹⁴
- 1 Pete xp⁶¹⁵
- 8 Peteturtle⁶¹⁶
- 1 Phaedriel the Vandal⁶¹⁷
- 1 Philipmac⁶¹⁸
- 3 PhilippWeissenbacher⁶¹⁹
- 3 Phonuz⁶²⁰
- 1 Physicsdude⁶²¹

597 <http://en.wikibooks.org/w/index.php?title=User:Othmansk2>
598 <http://en.wikibooks.org/w/index.php?title=User:Outbacksam34>
599 <http://en.wikibooks.org/w/index.php?title=User:Ownthor>
600 <http://en.wikibooks.org/w/index.php?title=User:Ozgrin>
601 <http://en.wikibooks.org/w/index.php?title=User:PKHG>
602 <http://en.wikibooks.org/w/index.php?title=User:POpZ>
603 <http://en.wikibooks.org/w/index.php?title=User:Pamtango>
604 <http://en.wikibooks.org/w/index.php?title=User:Panic2k4>
605 <http://en.wikibooks.org/w/index.php?title=User:Paperclip>
606 <http://en.wikibooks.org/w/index.php?title=User:ParallaxTZ>
607 <http://en.wikibooks.org/w/index.php?title=User:PartierSP>
608 <http://en.wikibooks.org/w/index.php?title=User:Patrolboat>
609 <http://en.wikibooks.org/w/index.php?title=User:PaulJC>
610 <http://en.wikibooks.org/w/index.php?title=User:Pavroo>
611 <http://en.wikibooks.org/w/index.php?title=User:Pearts>
612 http://en.wikibooks.org/w/index.php?title=User:Pedro_Fonini
613 <http://en.wikibooks.org/w/index.php?title=User:Pengo>
614 <http://en.wikibooks.org/w/index.php?title=User:Petariot>
615 http://en.wikibooks.org/w/index.php?title=User:Pete_xp
616 <http://en.wikibooks.org/w/index.php?title=User:Peteturtle>
617 http://en.wikibooks.org/w/index.php?title=User:Phaedriel_the_Vandal
618 <http://en.wikibooks.org/w/index.php?title=User:Philipmac>
619 <http://en.wikibooks.org/w/index.php?title=User:PhilippWeissenbacher>
620 <http://en.wikibooks.org/w/index.php?title=User:Phonuz>
621 <http://en.wikibooks.org/w/index.php?title=User:Physicsdude>

4	Pignol23 ⁶²²
1	Pikpus ⁶²³
1	Pinkmouse ⁶²⁴
1	Pinzo ⁶²⁵
2	Pixel Eater ⁶²⁶
1	Pixelstuff ⁶²⁷
1	Plasmasphere ⁶²⁸
1	Polluks ⁶²⁹
1	Pootworm ⁶³⁰
138	Popski ⁶³¹
3	Porlando ⁶³²
2	PowuaTeam ⁶³³
2	Pquijal ⁶³⁴
1	Priit K. ⁶³⁵
2	Professional noob ⁶³⁶
5	ProfoX ⁶³⁷
1	Prydain55 ⁶³⁸
4	Psychogears ⁶³⁹
4	Pwd ⁶⁴⁰
1	Pyro guy ⁶⁴¹
18	Qrc ⁶⁴²
3	Quantufinity@gmail.com ⁶⁴³
184	Quantum Anomaly ⁶⁴⁴
1	Quibik ⁶⁴⁵
27	QuiteUnusual ⁶⁴⁶

622	http://en.wikibooks.org/w/index.php?title=User:Pignol23
623	http://en.wikibooks.org/w/index.php?title=User:Pikpus
624	http://en.wikibooks.org/w/index.php?title=User:Pinkmouse
625	http://en.wikibooks.org/w/index.php?title=User:Pinzo
626	http://en.wikibooks.org/w/index.php?title=User:Pixel_Eater
627	http://en.wikibooks.org/w/index.php?title=User:Pixelstuff
628	http://en.wikibooks.org/w/index.php?title=User:Plasmasphere
629	http://en.wikibooks.org/w/index.php?title=User:Polluks
630	http://en.wikibooks.org/w/index.php?title=User:Pootworm
631	http://en.wikibooks.org/w/index.php?title=User:Popski
632	http://en.wikibooks.org/w/index.php?title=User:Porlando
633	http://en.wikibooks.org/w/index.php?title=User:PowuaTeam
634	http://en.wikibooks.org/w/index.php?title=User:Pquijal
635	http://en.wikibooks.org/w/index.php?title=User:Priit_K.
636	http://en.wikibooks.org/w/index.php?title=User:Professional_noob
637	http://en.wikibooks.org/w/index.php?title=User:ProfoX
638	http://en.wikibooks.org/w/index.php?title=User:Prydain55
639	http://en.wikibooks.org/w/index.php?title=User:Psychogears
640	http://en.wikibooks.org/w/index.php?title=User:Pwd
641	http://en.wikibooks.org/w/index.php?title=User:Pyro_guy
642	http://en.wikibooks.org/w/index.php?title=User:Qrc
643	http://en.wikibooks.org/w/index.php?title=User:Quantufinity%40gmail.com
644	http://en.wikibooks.org/w/index.php?title=User:Quantum_Anomaly
645	http://en.wikibooks.org/w/index.php?title=User:Quibik
646	http://en.wikibooks.org/w/index.php?title=User:QuiteUnusual

- 1 Qwertyzzz18⁶⁴⁷
- 2 RVRgeek⁶⁴⁸
- 2 Rabenschwinge⁶⁴⁹
- 51 Radialronnie⁶⁵⁰
- 2 Radiat-r⁶⁵¹
- 1 Radly⁶⁵²
- 1 Raicuandi⁶⁵³
- 1 Rajeshja⁶⁵⁴
- 11 Ramac⁶⁵⁵
- 3 Randolph Richardson⁶⁵⁶
- 2 Random9q⁶⁵⁷
- 4 Rant⁶⁵⁸
- 11 RavenWhitehawk⁶⁵⁹
- 1 Ravichandar84⁶⁶⁰
- 1 RaymondSutanto⁶⁶¹
- 2 Rct36⁶⁶²
- 14 Rdp⁶⁶³
- 1 ReKast⁶⁶⁴
- 1 Rea00cy⁶⁶⁵
- 1 Realdealmagic⁶⁶⁶
- 1 Rebol⁶⁶⁷
- 1 Recall⁶⁶⁸
- 7 Recent Runes⁶⁶⁹
- 1 Red4tribe⁶⁷⁰
- 3 Redconfetti⁶⁷¹

647 <http://en.wikibooks.org/w/index.php?title=User:Qwertyzzz18>
 648 <http://en.wikibooks.org/w/index.php?title=User:RVRgeek>
 649 <http://en.wikibooks.org/w/index.php?title=User:Rabenschwinge>
 650 <http://en.wikibooks.org/w/index.php?title=User:Radialronnie>
 651 <http://en.wikibooks.org/w/index.php?title=User:Radiat-r>
 652 <http://en.wikibooks.org/w/index.php?title=User:Radly>
 653 <http://en.wikibooks.org/w/index.php?title=User:Raicuandi>
 654 <http://en.wikibooks.org/w/index.php?title=User:Rajeshja>
 655 <http://en.wikibooks.org/w/index.php?title=User:Ramac>
 656 http://en.wikibooks.org/w/index.php?title=User:Randolf_Richardson
 657 <http://en.wikibooks.org/w/index.php?title=User:Random9q>
 658 <http://en.wikibooks.org/w/index.php?title=User:Rant>
 659 <http://en.wikibooks.org/w/index.php?title=User:RavenWhitehawk>
 660 <http://en.wikibooks.org/w/index.php?title=User:Ravichandar84>
 661 <http://en.wikibooks.org/w/index.php?title=User:RaymondSutanto>
 662 <http://en.wikibooks.org/w/index.php?title=User:Rct36>
 663 <http://en.wikibooks.org/w/index.php?title=User:Rdp>
 664 <http://en.wikibooks.org/w/index.php?title=User:ReKast>
 665 <http://en.wikibooks.org/w/index.php?title=User:Rea00cy>
 666 <http://en.wikibooks.org/w/index.php?title=User:Realdealmagic>
 667 <http://en.wikibooks.org/w/index.php?title=User:Rebol>
 668 <http://en.wikibooks.org/w/index.php?title=User:Recall>
 669 http://en.wikibooks.org/w/index.php?title=User:Recent_Runes
 670 <http://en.wikibooks.org/w/index.php?title=User:Red4tribe>
 671 <http://en.wikibooks.org/w/index.php?title=User:Redconfetti>

- 12 Reddogzip⁶⁷²
- 2 Redyoshi49q⁶⁷³
- 7 Reece⁶⁷⁴
- 2 Reilithion⁶⁷⁵
- 1 Rejiquar⁶⁷⁶
- 3 Rekov⁶⁷⁷
- 2 Remi⁶⁷⁸
- 1 Remi0o⁶⁷⁹
- 8 Reptiles extintos⁶⁸⁰
- 5 Resul4e⁶⁸¹
- 3 Reverzentative⁶⁸²
- 1 Reyk⁶⁸³
- 1 Rgriffogoes⁶⁸⁴
- 2 Rheinhardt2⁶⁸⁵
- 2 Rhettro⁶⁸⁶
- 2 Richard 27⁶⁸⁷
- 3 Rjh009⁶⁸⁸
- 1 Rltromble⁶⁸⁹
- 22 Rm5248⁶⁹⁰
- 1 Robfox70⁶⁹¹
- 1 Robsbots⁶⁹²
- 1 Robstafarian⁶⁹³
- 1 Robsterbertling⁶⁹⁴
- 1 RogerWickes⁶⁹⁵
- 2 Roli⁶⁹⁶

-
- 672 <http://en.wikibooks.org/w/index.php?title=User:Reddogzip>
 - 673 <http://en.wikibooks.org/w/index.php?title=User:Redyoshi49q>
 - 674 <http://en.wikibooks.org/w/index.php?title=User:Reece>
 - 675 <http://en.wikibooks.org/w/index.php?title=User:Reilithion>
 - 676 <http://en.wikibooks.org/w/index.php?title=User:Rejiquar>
 - 677 <http://en.wikibooks.org/w/index.php?title=User:Rekov>
 - 678 <http://en.wikibooks.org/w/index.php?title=User:Remi>
 - 679 <http://en.wikibooks.org/w/index.php?title=User:Remi0o>
 - 680 http://en.wikibooks.org/w/index.php?title=User:Reptiles_extintos
 - 681 <http://en.wikibooks.org/w/index.php?title=User:Resul4e>
 - 682 <http://en.wikibooks.org/w/index.php?title=User:Reverzentative>
 - 683 <http://en.wikibooks.org/w/index.php?title=User:Reyk>
 - 684 <http://en.wikibooks.org/w/index.php?title=User:Rgriffogoes>
 - 685 <http://en.wikibooks.org/w/index.php?title=User:Rheinhardt2>
 - 686 <http://en.wikibooks.org/w/index.php?title=User:Rhettro>
 - 687 http://en.wikibooks.org/w/index.php?title=User:Richard_27
 - 688 <http://en.wikibooks.org/w/index.php?title=User:Rjh009>
 - 689 <http://en.wikibooks.org/w/index.php?title=User:Rltromble>
 - 690 <http://en.wikibooks.org/w/index.php?title=User:Rm5248>
 - 691 <http://en.wikibooks.org/w/index.php?title=User:Robfox70>
 - 692 <http://en.wikibooks.org/w/index.php?title=User:Robsbots>
 - 693 <http://en.wikibooks.org/w/index.php?title=User:Robstafarian>
 - 694 <http://en.wikibooks.org/w/index.php?title=User:Robsterbertling>
 - 695 <http://en.wikibooks.org/w/index.php?title=User:RogerWickes>
 - 696 <http://en.wikibooks.org/w/index.php?title=User:Roli>

- 29 Romainbehar⁶⁹⁷
- 1 Ron333⁶⁹⁸
- 8 Root⁶⁹⁹
- 75 Rosver⁷⁰⁰
- 1 Roxorz42⁷⁰¹
- 11 Rpyle731⁷⁰²
- 1 Rrreutel⁷⁰³
- 2 Rukaru⁷⁰⁴
- 1 Ruvyen⁷⁰⁵
- 5 Rwst⁷⁰⁶
- 3 Rylz⁷⁰⁷
- 1 SBJohnny⁷⁰⁸
- 8 SETH HIKARU⁷⁰⁹
- 1 SK123r⁷¹⁰
- 2 Samnsparky⁷¹¹
- 1 Sandalle⁷¹²
- 1 Sandothegrate⁷¹³
- 5 Sandsmertz⁷¹⁴
- 1 Sankekur⁷¹⁵
- 4 Sapalskimichal⁷¹⁶
- 7 SatanClaus⁷¹⁷
- 1 Satribe101⁷¹⁸
- 1 Savage Beef⁷¹⁹
- 8 Saysaknow⁷²⁰
- 2 Scar⁷²¹

697 <http://en.wikibooks.org/w/index.php?title=User:Romainbehar>

698 <http://en.wikibooks.org/w/index.php?title=User:Ron333>

699 <http://en.wikibooks.org/w/index.php?title=User:Root>

700 <http://en.wikibooks.org/w/index.php?title=User:Rosver>

701 <http://en.wikibooks.org/w/index.php?title=User:Roxorz42>

702 <http://en.wikibooks.org/w/index.php?title=User:Rpyle731>

703 <http://en.wikibooks.org/w/index.php?title=User:Rrreutel>

704 <http://en.wikibooks.org/w/index.php?title=User:Rukaru>

705 <http://en.wikibooks.org/w/index.php?title=User:Ruvyen>

706 <http://en.wikibooks.org/w/index.php?title=User:Rwst>

707 <http://en.wikibooks.org/w/index.php?title=User:Rylz>

708 <http://en.wikibooks.org/w/index.php?title=User:SBJohnny>

709 http://en.wikibooks.org/w/index.php?title=User:SETH_HIKARU

710 <http://en.wikibooks.org/w/index.php?title=User:SK123r>

711 <http://en.wikibooks.org/w/index.php?title=User:Samnsparky>

712 <http://en.wikibooks.org/w/index.php?title=User:Sandalle>

713 <http://en.wikibooks.org/w/index.php?title=User:Sandothegrate>

714 <http://en.wikibooks.org/w/index.php?title=User:Sandsmertz>

715 <http://en.wikibooks.org/w/index.php?title=User:Sankekur>

716 <http://en.wikibooks.org/w/index.php?title=User:Sapalskimichal>

717 <http://en.wikibooks.org/w/index.php?title=User:SatanClaus>

718 <http://en.wikibooks.org/w/index.php?title=User:Satribe101>

719 http://en.wikibooks.org/w/index.php?title=User:Savage_Beef

720 <http://en.wikibooks.org/w/index.php?title=User:Saysaknow>

721 <http://en.wikibooks.org/w/index.php?title=User:Scar>

4	ScaroDj ⁷²²
32	SchmS ⁷²³
1	Schultz.Ryan ⁷²⁴
1	Sckoobs ⁷²⁵
1	Scottpledger ⁷²⁶
1	Sean.flynn ⁷²⁷
3	Selfhiding ⁷²⁸
1	Selkovjr ⁷²⁹
5	Sendoshin ⁷³⁰
1	Sephlaire ⁷³¹
1	Seraphrean ⁷³²
55	SergeantOreo ⁷³³
3	Serpentine Cougar ⁷³⁴
8	Sestina ⁷³⁵
2	Sesu Prime ⁷³⁶
2	Sevcsik ⁷³⁷
13	SgtBursk ⁷³⁸
3	ShadoCrytr ⁷³⁹
1	Shadowimage ⁷⁴⁰
1	Shellrie ⁷⁴¹
1	Sheridan ⁷⁴²
2	Shinjin ⁷⁴³
7	Shinmoos ⁷⁴⁴
12	Shinydarkrai94 ⁷⁴⁵
2	Shokuku ⁷⁴⁶

722 <http://en.wikibooks.org/w/index.php?title=User:ScaroDj>

723 <http://en.wikibooks.org/w/index.php?title=User:SchmS>

724 <http://en.wikibooks.org/w/index.php?title=User:Schultz.Ryan>

725 <http://en.wikibooks.org/w/index.php?title=User:Sckoobs>

726 <http://en.wikibooks.org/w/index.php?title=User:Scottpledger>

727 <http://en.wikibooks.org/w/index.php?title=User:Sean.flynn>

728 <http://en.wikibooks.org/w/index.php?title=User:Selfhiding>

729 <http://en.wikibooks.org/w/index.php?title=User:Selkovjr>

730 <http://en.wikibooks.org/w/index.php?title=User:Sendoshin>

731 <http://en.wikibooks.org/w/index.php?title=User:Sephlaire>

732 <http://en.wikibooks.org/w/index.php?title=User:Seraphrean>

733 <http://en.wikibooks.org/w/index.php?title=User:SergeantOreo>

734 http://en.wikibooks.org/w/index.php?title=User:Serpentine_Cougar

735 <http://en.wikibooks.org/w/index.php?title=User:Sestina>

736 http://en.wikibooks.org/w/index.php?title=User:Sesu_Prime

737 <http://en.wikibooks.org/w/index.php?title=User:Sevcsik>

738 <http://en.wikibooks.org/w/index.php?title=User:SgtBursk>

739 <http://en.wikibooks.org/w/index.php?title=User:ShadoCrytr>

740 <http://en.wikibooks.org/w/index.php?title=User:Shadowimage>

741 <http://en.wikibooks.org/w/index.php?title=User:Shellrie>

742 <http://en.wikibooks.org/w/index.php?title=User:Sheridan>

743 <http://en.wikibooks.org/w/index.php?title=User:Shinjin>

744 <http://en.wikibooks.org/w/index.php?title=User:Shinmoos>

745 <http://en.wikibooks.org/w/index.php?title=User:Shinydarkrai94>

746 <http://en.wikibooks.org/w/index.php?title=User:Shokuku>

39	Shythinker ⁷⁴⁷
2	Siesta ⁷⁴⁸
2	Siggimund ⁷⁴⁹
8	Sigma ⁷⁵⁰
1	Simeon ⁷⁵¹
1	Simon W ⁷⁵²
1	Sinistra D32 ⁷⁵³
4	SixFt12 ⁷⁵⁴
1	Sjc ⁷⁵⁵
1	Skaruts ⁷⁵⁶
1	Skatanist ⁷⁵⁷
3	Skeep ⁷⁵⁸
2	Sketlover ⁷⁵⁹
1	Skirrid ⁷⁶⁰
4	Slady ⁷⁶¹
6	Slashme ⁷⁶²
11	Slayerjp ⁷⁶³
1	Sledz41 ⁷⁶⁴
1	Slush15player ⁷⁶⁵
1	SmileyCactus ⁷⁶⁶
2	Snarius ⁷⁶⁷
2	Socratesone ⁷⁶⁸
44	Soeb ⁷⁶⁹
2	Someguywithnothingtodo ⁷⁷⁰
1	Sonic4Spuds ⁷⁷¹

⁷⁴⁷ <http://en.wikibooks.org/w/index.php?title=User:Shythinker>

⁷⁴⁸ <http://en.wikibooks.org/w/index.php?title=User:Siesta>

⁷⁴⁹ <http://en.wikibooks.org/w/index.php?title=User:Siggimund>

⁷⁵⁰ http://en.wikibooks.org/w/index.php?title=User:Sigma_7

⁷⁵¹ <http://en.wikibooks.org/w/index.php?title=User:Simeon>

⁷⁵² http://en.wikibooks.org/w/index.php?title=User:Simon_W

⁷⁵³ http://en.wikibooks.org/w/index.php?title=User:Sinistra_D32

⁷⁵⁴ <http://en.wikibooks.org/w/index.php?title=User:SixFt12>

⁷⁵⁵ <http://en.wikibooks.org/w/index.php?title=User:Sjc>

⁷⁵⁶ <http://en.wikibooks.org/w/index.php?title=User:Skaruts>

⁷⁵⁷ <http://en.wikibooks.org/w/index.php?title=User:Skatanist>

⁷⁵⁸ <http://en.wikibooks.org/w/index.php?title=User:Skeep>

⁷⁵⁹ <http://en.wikibooks.org/w/index.php?title=User:Sketlover>

⁷⁶⁰ <http://en.wikibooks.org/w/index.php?title=User:Skirrid>

⁷⁶¹ <http://en.wikibooks.org/w/index.php?title=User:Slady>

⁷⁶² <http://en.wikibooks.org/w/index.php?title=User:Slashme>

⁷⁶³ <http://en.wikibooks.org/w/index.php?title=User:Slayerjp>

⁷⁶⁴ <http://en.wikibooks.org/w/index.php?title=User:Sledz41>

⁷⁶⁵ <http://en.wikibooks.org/w/index.php?title=User:Slush15player>

⁷⁶⁶ <http://en.wikibooks.org/w/index.php?title=User:SmileyCactus>

⁷⁶⁷ <http://en.wikibooks.org/w/index.php?title=User:Snarius>

⁷⁶⁸ <http://en.wikibooks.org/w/index.php?title=User:Socratesone>

⁷⁶⁹ <http://en.wikibooks.org/w/index.php?title=User:Soeb>

⁷⁷⁰ <http://en.wikibooks.org/w/index.php?title=User:Someguywithnothingtodo>

⁷⁷¹ <http://en.wikibooks.org/w/index.php?title=User:Sonic4Spuds>

2	Soriyath ⁷⁷²
146	SoylentGreen ⁷⁷³
12	Soylentgreen ⁷⁷⁴
2	Spartacus3d ⁷⁷⁵
10	Spellmaker ⁷⁷⁶
451	Spiderworm ⁷⁷⁷
3	Spiffyandy ⁷⁷⁸
4	Spongebob88 ⁷⁷⁹
3	Squaremo ⁷⁸⁰
14	Stalepanda ⁷⁸¹
1	Stefantalpalaru ⁷⁸²
938	Stepheng3 ⁷⁸³
4	Sterre ⁷⁸⁴
2	Stormy ⁷⁸⁵
1	SumOfwHoly ⁷⁸⁶
15	Sumoncil ⁷⁸⁷
1	Sunny256 ⁷⁸⁸
1	Super Dave ⁷⁸⁹
5	Super3boy ⁷⁹⁰
1	Swap i ⁷⁹¹
3	Swift ⁷⁹²
2	T.C.Fire ⁷⁹³
27	TBOL3 ⁷⁹⁴
3	Tal500 ⁷⁹⁵
56	Tannersf ⁷⁹⁶

772 <http://en.wikibooks.org/w/index.php?title=User:Soriyath>
773 <http://en.wikibooks.org/w/index.php?title=User:SoylentGreen>
774 <http://en.wikibooks.org/w/index.php?title=User:Soylentgreen>
775 <http://en.wikibooks.org/w/index.php?title=User:Spartacus3d>
776 <http://en.wikibooks.org/w/index.php?title=User:Spellmaker>
777 <http://en.wikibooks.org/w/index.php?title=User:Spiderworm>
778 <http://en.wikibooks.org/w/index.php?title=User:Spiffyandy>
779 <http://en.wikibooks.org/w/index.php?title=User:Spongebob88>
780 <http://en.wikibooks.org/w/index.php?title=User:Squaremo>
781 <http://en.wikibooks.org/w/index.php?title=User:Stalepanda>
782 <http://en.wikibooks.org/w/index.php?title=User:Stefantalpalaru>
783 <http://en.wikibooks.org/w/index.php?title=User:Stepheng3>
784 <http://en.wikibooks.org/w/index.php?title=User:Sterre>
785 <http://en.wikibooks.org/w/index.php?title=User:Stormy>
786 <http://en.wikibooks.org/w/index.php?title=User:SumOfwHoly>
787 <http://en.wikibooks.org/w/index.php?title=User:Sumoncil>
788 <http://en.wikibooks.org/w/index.php?title=User:Sunny256>
789 http://en.wikibooks.org/w/index.php?title=User:Super_Dave
790 <http://en.wikibooks.org/w/index.php?title=User:Super3boy>
791 http://en.wikibooks.org/w/index.php?title=User:Swap_i
792 <http://en.wikibooks.org/w/index.php?title=User:Swift>
793 <http://en.wikibooks.org/w/index.php?title=User:T.C.Fire>
794 <http://en.wikibooks.org/w/index.php?title=User:TBOL3>
795 <http://en.wikibooks.org/w/index.php?title=User:Tal500>
796 <http://en.wikibooks.org/w/index.php?title=User:Tannersf>

2	Tarot ⁷⁹⁷
2	Taye ⁷⁹⁸
2	Tayewiki ⁷⁹⁹
2	Tdunn7 ⁸⁰⁰
12	Teddygage ⁸⁰¹
4	Tedesson ⁸⁰²
1	Tedka ⁸⁰³
1	Telmer6 ⁸⁰⁴
2	Terrywallwork ⁸⁰⁵
1	Thangalin ⁸⁰⁶
1	That.1.guy ⁸⁰⁷
1	The enemies of god ⁸⁰⁸
11	Thenub314 ⁸⁰⁹
1	Thesandlord ⁸¹⁰
1	Thespeedylife ⁸¹¹
1	Theteach ⁸¹²
2	Thinkingman ⁸¹³
2	ThomasB ⁸¹⁴
7	ThomasShepard ⁸¹⁵
11	ThomasTenCate ⁸¹⁶
2	Thorack ⁸¹⁷
2	ThriiDæ ⁸¹⁸
2	Thud ⁸¹⁹
4	Thunder ⁸²⁰
2	Thzi ⁸²¹

797 <http://en.wikibooks.org/w/index.php?title=User:Tarot>
798 <http://en.wikibooks.org/w/index.php?title=User:Taye>
799 <http://en.wikibooks.org/w/index.php?title=User:Tayewiki>
800 <http://en.wikibooks.org/w/index.php?title=User:Tdunn7>
801 <http://en.wikibooks.org/w/index.php?title=User:Teddygage>
802 <http://en.wikibooks.org/w/index.php?title=User:Tedesson>
803 <http://en.wikibooks.org/w/index.php?title=User:Tedka>
804 <http://en.wikibooks.org/w/index.php?title=User:Telmer6>
805 <http://en.wikibooks.org/w/index.php?title=User:Terrywallwork>
806 <http://en.wikibooks.org/w/index.php?title=User:Thangalin>
807 <http://en.wikibooks.org/w/index.php?title=User:That.1.guy>
808 http://en.wikibooks.org/w/index.php?title=User:The_enemies_of_god
809 <http://en.wikibooks.org/w/index.php?title=User:Thenub314>
810 <http://en.wikibooks.org/w/index.php?title=User:Thesandlord>
811 <http://en.wikibooks.org/w/index.php?title=User:Thespeedylife>
812 <http://en.wikibooks.org/w/index.php?title=User:Theteach>
813 <http://en.wikibooks.org/w/index.php?title=User:Thinkingman>
814 <http://en.wikibooks.org/w/index.php?title=User:ThomasB>
815 <http://en.wikibooks.org/w/index.php?title=User:ThomasShepard>
816 <http://en.wikibooks.org/w/index.php?title=User:ThomasTenCate>
817 <http://en.wikibooks.org/w/index.php?title=User:Thorack>
818 <http://en.wikibooks.org/w/index.php?title=User:ThriiD%C3%A6>
819 <http://en.wikibooks.org/w/index.php?title=User:Thud>
820 <http://en.wikibooks.org/w/index.php?title=User:Thunder>
821 <http://en.wikibooks.org/w/index.php?title=User:Thzi>

- 1 TiMoch⁸²²
- 5 Tictoon⁸²³
- 2 Tigerstickfigure⁸²⁴
- 4 TimDumol⁸²⁵
- 9 Tjb0607⁸²⁶
- 1 Tmandry⁸²⁷
- 2 Tmr232⁸²⁸
- 9 Tom soderlund⁸²⁹
- 1 Tommciver⁸³⁰
- 1 Tommy g003⁸³¹
- 6 Tosnic⁸³²
- 1 Totalwriter0331⁸³³
- 5 Totophe64⁸³⁴
- 1 Tpower⁸³⁵
- 1 Trezero⁸³⁶
- 6 Trident⁸³⁷
- 1 Tsduv21⁸³⁸
- 3 Tualha⁸³⁹
- 89 Turtle Man⁸⁴⁰
- 16 Twofingers⁸⁴¹
- 12 Tyaedalis⁸⁴²
- 2 Tybee⁸⁴³
- 2 Tyrant monkey⁸⁴⁴
- 2 Ubai⁸⁴⁵
- 1 Uncle G⁸⁴⁶

822 <http://en.wikibooks.org/w/index.php?title=User:TiMoch>
823 <http://en.wikibooks.org/w/index.php?title=User:Tictoon>
824 <http://en.wikibooks.org/w/index.php?title=User:Tigerstickfigure>
825 <http://en.wikibooks.org/w/index.php?title=User:TimDumol>
826 <http://en.wikibooks.org/w/index.php?title=User:Tjb0607>
827 <http://en.wikibooks.org/w/index.php?title=User:Tmandry>
828 <http://en.wikibooks.org/w/index.php?title=User:Tmr232>
829 http://en.wikibooks.org/w/index.php?title=User:Tom_soderlund
830 <http://en.wikibooks.org/w/index.php?title=User:Tommciver>
831 http://en.wikibooks.org/w/index.php?title=User:Tommy_g003
832 <http://en.wikibooks.org/w/index.php?title=User:Tosnic>
833 <http://en.wikibooks.org/w/index.php?title=User:Totalwriter0331>
834 <http://en.wikibooks.org/w/index.php?title=User:Totophe64>
835 <http://en.wikibooks.org/w/index.php?title=User:Tpower>
836 <http://en.wikibooks.org/w/index.php?title=User:Trezero>
837 <http://en.wikibooks.org/w/index.php?title=User:Trident>
838 <http://en.wikibooks.org/w/index.php?title=User:Tsduv21>
839 <http://en.wikibooks.org/w/index.php?title=User:Tualha>
840 http://en.wikibooks.org/w/index.php?title=User:Turtle_Man
841 <http://en.wikibooks.org/w/index.php?title=User:Twofingers>
842 <http://en.wikibooks.org/w/index.php?title=User:Tyaedalis>
843 <http://en.wikibooks.org/w/index.php?title=User:Tybee>
844 http://en.wikibooks.org/w/index.php?title=User:Tyrant_monkey
845 <http://en.wikibooks.org/w/index.php?title=User:Ubai>
846 http://en.wikibooks.org/w/index.php?title=User:Uncle_G

- 2 UnknownGuy⁸⁴⁷
- 1 UsedHONDA⁸⁴⁸
- 3 VB4231⁸⁴⁹
- 2 Vaakmeisster⁸⁵⁰
- 1 Valdemarasl⁸⁵¹
- 1 Valid User⁸⁵²
- 1 Van der Hoorn⁸⁵³
- 1 Vapor⁸⁵⁴
- 1 Vault⁸⁵⁵
- 1 Vaxine19⁸⁵⁶
- 2 Velichi⁸⁵⁷
- 2 Velociostrich⁸⁵⁸
- 18 Venar303⁸⁵⁹
- 1 Vi3gamehkr⁸⁶⁰
- 1 Vrabec⁸⁶¹
- 2 Waximo⁸⁶²
- 5 Webaware⁸⁶³
- 10 Weeix⁸⁶⁴
- 4 WeirdHat⁸⁶⁵
- 5 WesleyJL⁸⁶⁶
- 1 Wgoelkel⁸⁶⁷
- 1 WiTr⁸⁶⁸
- 1 Wikibooks is Communism⁸⁶⁹
- 2 Wikidieter⁸⁷⁰
- 2 Wildern⁸⁷¹

847 <http://en.wikibooks.org/w/index.php?title=User:UnknownGuy>

848 <http://en.wikibooks.org/w/index.php?title=User:UsedHONDA>

849 <http://en.wikibooks.org/w/index.php?title=User:VB4231>

850 <http://en.wikibooks.org/w/index.php?title=User:Vaakmeisster>

851 <http://en.wikibooks.org/w/index.php?title=User:Valdemarasl>

852 http://en.wikibooks.org/w/index.php?title=User:Valid_User

853 http://en.wikibooks.org/w/index.php?title=User:Van_der_Hoorn

854 <http://en.wikibooks.org/w/index.php?title=User:Vapor>

855 <http://en.wikibooks.org/w/index.php?title=User:Vault>

856 <http://en.wikibooks.org/w/index.php?title=User:Vaxine19>

857 <http://en.wikibooks.org/w/index.php?title=User:Velichi>

858 <http://en.wikibooks.org/w/index.php?title=User:Velociostrich>

859 <http://en.wikibooks.org/w/index.php?title=User:Venar303>

860 <http://en.wikibooks.org/w/index.php?title=User:Vi3gamehkr>

861 <http://en.wikibooks.org/w/index.php?title=User:Vrabec>

862 <http://en.wikibooks.org/w/index.php?title=User:Waximo>

863 <http://en.wikibooks.org/w/index.php?title=User:Webaware>

864 <http://en.wikibooks.org/w/index.php?title=User:Weeix>

865 <http://en.wikibooks.org/w/index.php?title=User:WeirdHat>

866 <http://en.wikibooks.org/w/index.php?title=User:WesleyJL>

867 <http://en.wikibooks.org/w/index.php?title=User:Wgoelkel>

868 <http://en.wikibooks.org/w/index.php?title=User:WiTr>

869 http://en.wikibooks.org/w/index.php?title=User:Wikibooks_is_Communism

870 <http://en.wikibooks.org/w/index.php?title=User:Wikidieter>

871 <http://en.wikibooks.org/w/index.php?title=User:Wildern>

- 1 Willemd⁸⁷²
- 1 Wimjongman⁸⁷³
- 3 WindsorSpring⁸⁷⁴
- 11 Winterbourne⁸⁷⁵
- 2 Winters⁸⁷⁶
- 18 WithAlligators⁸⁷⁷
- 6 Withinfocus⁸⁷⁸
- 1 WolfeBay⁸⁷⁹
- 1 WolfprintFX⁸⁸⁰
- 1 Wolvenmoon⁸⁸¹
- 1 Woodsjay⁸⁸²
- 3 Woodstocket⁸⁸³
- 1 Worm 19⁸⁸⁴
- 1 Wwwwolf⁸⁸⁵
- 51 Wynnmc⁸⁸⁶
- 12 XDraconian⁸⁸⁷
- 2 Xander98989⁸⁸⁸
- 15 Xania⁸⁸⁹
- 1 Xemkis12⁸⁹⁰
- 2 Xerol⁸⁹¹
- 1 Xiephoire⁸⁹²
- 5 Xlinuxkdex⁸⁹³
- 2 Xwriter⁸⁹⁴
- 5 YMS⁸⁹⁵
- 2 Yashnz⁸⁹⁶

872 <http://en.wikibooks.org/w/index.php?title=User:Willemd>
873 <http://en.wikibooks.org/w/index.php?title=User:Wimjongman>
874 <http://en.wikibooks.org/w/index.php?title=User:WindsorSpring>
875 <http://en.wikibooks.org/w/index.php?title=User:Winterbourne>
876 <http://en.wikibooks.org/w/index.php?title=User:Winters>
877 <http://en.wikibooks.org/w/index.php?title=User:WithAlligators>
878 <http://en.wikibooks.org/w/index.php?title=User:Withinfocus>
879 <http://en.wikibooks.org/w/index.php?title=User:WolfeBay>
880 <http://en.wikibooks.org/w/index.php?title=User:WolfprintFX>
881 <http://en.wikibooks.org/w/index.php?title=User:Wolvenmoon>
882 <http://en.wikibooks.org/w/index.php?title=User:Woodsjay>
883 <http://en.wikibooks.org/w/index.php?title=User:Woodstocket>
884 http://en.wikibooks.org/w/index.php?title=User:Worm_19
885 <http://en.wikibooks.org/w/index.php?title=User:Wwwwolf>
886 <http://en.wikibooks.org/w/index.php?title=User:Wynnmc>
887 <http://en.wikibooks.org/w/index.php?title=User:XDraconian>
888 <http://en.wikibooks.org/w/index.php?title=User:Xander98989>
889 <http://en.wikibooks.org/w/index.php?title=User:Xania>
890 <http://en.wikibooks.org/w/index.php?title=User:Xemkis12>
891 <http://en.wikibooks.org/w/index.php?title=User:Xerol>
892 <http://en.wikibooks.org/w/index.php?title=User:Xiephoire>
893 <http://en.wikibooks.org/w/index.php?title=User:Xlinuxkdex>
894 <http://en.wikibooks.org/w/index.php?title=User:Xwriter>
895 <http://en.wikibooks.org/w/index.php?title=User:YMS>
896 <http://en.wikibooks.org/w/index.php?title=User:Yashnz>

- 2 Yes Man⁸⁹⁷
- 3 YesPoX⁸⁹⁸
- 13 Yesuto15⁸⁹⁹
- 4 Yewbow⁹⁰⁰
- 2 Yhevhe⁹⁰¹
- 1 Yhipavahit⁹⁰²
- 2 Yjvyas⁹⁰³
- 1 Yosefm⁹⁰⁴
- 7 Yoshi1476⁹⁰⁵
- 18 Yosofun⁹⁰⁶
- 1 Yoyotweak⁹⁰⁷
- 1 Ysangkok⁹⁰⁸
- 1 Yuyuyak⁹⁰⁹
- 1 Zachary Murray⁹¹⁰
- 2 Zarathoustra⁹¹¹
- 1 Zbrown⁹¹²
- 27 ZeroOne⁹¹³
- 8 Zeviion⁹¹⁴
- 16 Zilch⁹¹⁵
- 1 Zilla⁹¹⁶
- 5 Zoohouse⁹¹⁷
- 1 Δ⁹¹⁸

-
- 897 http://en.wikibooks.org/w/index.php?title=User:Yes_Man
 - 898 <http://en.wikibooks.org/w/index.php?title=User:YesPoX>
 - 899 <http://en.wikibooks.org/w/index.php?title=User:Yesuto15>
 - 900 <http://en.wikibooks.org/w/index.php?title=User:Yewbow>
 - 901 <http://en.wikibooks.org/w/index.php?title=User:Yhevhe>
 - 902 <http://en.wikibooks.org/w/index.php?title=User:Yhipavahit>
 - 903 <http://en.wikibooks.org/w/index.php?title=User:Yjvyas>
 - 904 <http://en.wikibooks.org/w/index.php?title=User:Yosefm>
 - 905 <http://en.wikibooks.org/w/index.php?title=User:Yoshi1476>
 - 906 <http://en.wikibooks.org/w/index.php?title=User:Yosofun>
 - 907 <http://en.wikibooks.org/w/index.php?title=User:Yoyotweak>
 - 908 <http://en.wikibooks.org/w/index.php?title=User:Ysangkok>
 - 909 <http://en.wikibooks.org/w/index.php?title=User:Yuyuyak>
 - 910 http://en.wikibooks.org/w/index.php?title=User:Zachary_Murray
 - 911 <http://en.wikibooks.org/w/index.php?title=User:Zarathoustra>
 - 912 <http://en.wikibooks.org/w/index.php?title=User:Zbrown>
 - 913 <http://en.wikibooks.org/w/index.php?title=User:ZeroOne>
 - 914 <http://en.wikibooks.org/w/index.php?title=User:Zeviion>
 - 915 <http://en.wikibooks.org/w/index.php?title=User:Zilch>
 - 916 <http://en.wikibooks.org/w/index.php?title=User:Zilla>
 - 917 <http://en.wikibooks.org/w/index.php?title=User:Zoohouse>
 - 918 <http://en.wikibooks.org/w/index.php?title=User:%CE%94>

List of Figures

- GFDL: Gnu Free Documentation License. <http://www.gnu.org/licenses/fdl.html>
- cc-by-sa-3.0: Creative Commons Attribution ShareAlike 3.0 License. <http://creativecommons.org/licenses/by-sa/3.0/>
- cc-by-sa-2.5: Creative Commons Attribution ShareAlike 2.5 License. <http://creativecommons.org/licenses/by-sa/2.5/>
- cc-by-sa-2.0: Creative Commons Attribution ShareAlike 2.0 License. <http://creativecommons.org/licenses/by-sa/2.0/>
- cc-by-sa-1.0: Creative Commons Attribution ShareAlike 1.0 License. <http://creativecommons.org/licenses/by-sa/1.0/>
- cc-by-2.0: Creative Commons Attribution 2.0 License. <http://creativecommons.org/licenses/by/2.0/>
- cc-by-2.0: Creative Commons Attribution 2.0 License. <http://creativecommons.org/licenses/by/2.0/deed.en>
- cc-by-2.5: Creative Commons Attribution 2.5 License. <http://creativecommons.org/licenses/by/2.5/deed.en>
- cc-by-3.0: Creative Commons Attribution 3.0 License. <http://creativecommons.org/licenses/by/3.0/deed.en>
- GPL: GNU General Public License. <http://www.gnu.org/licenses/gpl-2.0.txt>
- LGPL: GNU Lesser General Public License. <http://www.gnu.org/licenses/lgpl.html>
- PD: This image is in the public domain.
- ATTR: The copyright holder of this file allows anyone to use it for any purpose, provided that the copyright holder is properly attributed. Redistribution, derivative work, commercial use, and all other use is permitted.
- EURO: This is the common (reverse) face of a euro coin. The copyright on the design of the common face of the euro coins belongs to the European Commission. Authorised is reproduction in a format without relief (drawings, paintings, films) provided they are not detrimental to the image of the euro.
- LFK: Lizenz Freie Kunst. <http://artlibre.org/licence/lal/de>
- CFR: Copyright free use.

- EPL: Eclipse Public License. <http://www.eclipse.org/org/documents/epl-v10.php>

Copies of the GPL, the LGPL as well as a GFDL are included in chapter Licenses⁹¹⁹. Please note that images in the public domain do not require attribution. You may click on the image numbers in the following table to open the webpage of the images in your webbrowser.

919 Chapter 244 on page 1851

1	Michael Otto (user Mayqel ⁹²⁰)	
2	Ldo ⁹²¹	cc-by-sa-3.0
3	Abdull ⁹²²	GFDL
4	Ldo ⁹²³	cc-by-sa-3.0
5	SoylentGreen ⁹²⁴	PD
6	Ldo ⁹²⁵	cc-by-sa-3.0
7	Ldo ⁹²⁶	cc-by-sa-3.0
8	Ldo ⁹²⁷	cc-by-sa-3.0
9	Ldo ⁹²⁸	cc-by-sa-3.0
10	Ldo ⁹²⁹	cc-by-sa-3.0
11	Ldo ⁹³⁰	cc-by-sa-3.0
12	Ldo ⁹³¹	cc-by-sa-3.0
13	Ldo ⁹³²	cc-by-sa-3.0
14	Ldo ⁹³³	cc-by-sa-3.0
15	Ldo ⁹³⁴	cc-by-sa-3.0
16	Original uploader was BANZ111, at en.wikibooks ⁹³⁵	PD
17	b:en:User:BANZ111 ⁹³⁶	PD
18	Marco Polo ⁹³⁷ at en.wikipedia ⁹³⁸	PD
19	Marco Polo ⁹³⁹ at en.wikipedia ⁹⁴⁰	PD
20	Marco Polo ⁹⁴¹ at en.wikipedia ⁹⁴²	PD
21	SoylentGreen ⁹⁴³	GFDL
22	SoylentGreen ⁹⁴⁴	GFDL
23	SoylentGreen ⁹⁴⁵	GFDL
24	SoylentGreen ⁹⁴⁶	GFDL

- [920 http://en.wikibooks.org/wiki/User%3AMayqel](http://en.wikibooks.org/wiki/User%3AMayqel)
[921 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[922 http://en.wikibooks.org/wiki/User%3AAbdull](http://en.wikibooks.org/wiki/User%3AAbdull)
[923 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[924 http://en.wikibooks.org/wiki/User%3ASoylentGreen](http://en.wikibooks.org/wiki/User%3ASoylentGreen)
[925 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[926 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[927 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[928 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[929 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[930 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[931 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[932 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[933 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[934 http://en.wikibooks.org/wiki/User%3ALdo](http://en.wikibooks.org/wiki/User%3ALdo)
[935 http://en.wikibooks.org](http://en.wikibooks.org)
[936 http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ABANZ111](http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ABANZ111)
[937 http://en.wikibooks.org/wiki/%3Aen%3AUser%3AMarco%20Polo](http://en.wikibooks.org/wiki/%3Aen%3AUser%3AMarco%20Polo)
[938 http://en.wikipedia.org](http://en.wikipedia.org)
[939 http://en.wikibooks.org/wiki/%3Aen%3AUser%3AMarco%20Polo](http://en.wikibooks.org/wiki/%3Aen%3AUser%3AMarco%20Polo)
[940 http://en.wikipedia.org](http://en.wikipedia.org)
[941 http://en.wikibooks.org/wiki/%3Aen%3AUser%3AMarco%20Polo](http://en.wikibooks.org/wiki/%3Aen%3AUser%3AMarco%20Polo)
[942 http://en.wikipedia.org](http://en.wikipedia.org)
[943 http://en.wikibooks.org/wiki/User%3ASoylentGreen](http://en.wikibooks.org/wiki/User%3ASoylentGreen)
[944 http://en.wikibooks.org/wiki/User%3ASoylentGreen](http://en.wikibooks.org/wiki/User%3ASoylentGreen)
[945 http://en.wikibooks.org/wiki/User%3ASoylentGreen](http://en.wikibooks.org/wiki/User%3ASoylentGreen)
[946 http://en.wikibooks.org/wiki/User%3ASoylentGreen](http://en.wikibooks.org/wiki/User%3ASoylentGreen)

25	SoylentGreen ⁹⁴⁷	GFDL
26		cc-by-sa-2.0
27	Original uploader was Soeb ⁹⁴⁸ at en.wikibooks ⁹⁴⁹	
28	JamesNZ ⁹⁵⁰	
29	Rltromble ⁹⁵¹	cc-by-sa-3.0
30	JamesNZ (JamesNZ ⁹⁵² (talk ⁹⁵³))	
31	Blender Foundation	GPL
32	Blender Foundation	GPL
33	Blender Foundation	GPL
34	Blender Foundation	GPL
35	Blender Foundation	GPL
36	Blender Foundation	GPL
37	Blender Foundation	GPL
38	b:en>User:Spiderworm ⁹⁵⁴	PD
39	b:en>User:Spiderworm ⁹⁵⁵	PD
40		GPL
41		GPL
42	Blender Foundation	GPL
43	Blender Foundation	GPL
44		GPL
45	Blender Foundation	GPL
46	JamesNZ ⁹⁵⁶	
47	Blender Foundation	GPL
48	Blender Foundation	GPL
49	Blender Foundation	GPL
50	Blender Foundation	GPL
51	Blender Foundation	GPL
52	Blender Foundation	GPL
53	Blender Foundation	GPL
54	Blender Foundation	GPL
55	Blender Foundation	GPL
56	Original uploader was Lazy-lump ⁹⁵⁷ at en.wikibooks ⁹⁵⁸	PD
57	Blender Foundation	GPL
58	Lawrence D'Oliveiro	GPL
59	Lawrence D'Oliveiro	GPL
60	Lawrence D'Oliveiro	GPL
61	Lawrence D'Oliveiro	GPL
62		GPL
63		GPL

⁹⁴⁷ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

⁹⁴⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

⁹⁴⁹ <http://en.wikibooks.org>

⁹⁵⁰ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁵¹ <http://en.wikibooks.org/wiki/User%3ARltromble>

⁹⁵² <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁵³ <http://en.wikibooks.org/wiki/User%20talk%3AJamesNZ>

⁹⁵⁴ <http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASpiderworm>

⁹⁵⁵ <http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASpiderworm>

⁹⁵⁶ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁵⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

⁹⁵⁸ <http://en.wikibooks.org>

64		GPL
65	Lawrence D'Oliveiro	GPL
66	Lawrence D'Oliveiro	GPL
67		GPL
68	Vi3gamehkr ⁹⁵⁹	
69	Vi3gamehkr ⁹⁶⁰	
70	Vi3gamehkr ⁹⁶¹	
71		GPL
72		GPL
73	Blender Foundation ⁹⁶²	GPL
74	Blender Foundation ⁹⁶³	GPL
75	Blender Foundation ⁹⁶⁴	GPL
76	Rurik	PD
77	model by the Yo Frankie! team (see http://www.yofrankie.org/team/); ⁹⁶⁵ rendered by User:Stepheng3	
78	JamesNZ ⁹⁶⁶	
79	JamesNZ ⁹⁶⁷	
80	Blender Foundation	GPL
81		GPL
82	Blender Foundation	GPL
83	Blender Foundation	GPL
84	Blender Foundation	GPL
85	JamesNZ ⁹⁶⁸	
86		GPL
87	Blender Foundation	GPL
88	Blender Foundation	GPL
89	Blender Foundation	GPL
90	Blender Foundation	GPL
91	Blender Foundation	GPL
92	Blender Foundation	GPL
93	Blender Foundation	GPL
94	Ldo ⁹⁶⁹	
95	Ldo ⁹⁷⁰	
96	Blender Foundation	GPL
97	Blender Foundation	GPL
98	Blender Foundation	GPL
99	PvtKing ⁹⁷¹	cc-by-sa-3.0
100		GPL

959 <http://en.wikibooks.org/wiki/User%3AVi3gamehkr>

960 <http://en.wikibooks.org/wiki/User%3AVi3gamehkr>

961 <http://en.wikibooks.org/wiki/User%3AVi3gamehkr>

962 <http://en.wikibooks.org/wiki/Blender%20Foundation>

963 <http://en.wikibooks.org/wiki/Blender%20Foundation>

964 <http://en.wikibooks.org/wiki/Blender%20Foundation>

965 <http://www.yofrankie.org/team/>;

966 <http://en.wikibooks.org/wiki/User%3AJamesNZ>

967 <http://en.wikibooks.org/wiki/User%3AJamesNZ>

968 <http://en.wikibooks.org/wiki/User%3AJamesNZ>

969 <http://en.wikibooks.org/wiki/User%3ALdo>

970 <http://en.wikibooks.org/wiki/User%3ALdo>

971 <http://en.wikibooks.org/wiki/User%3APvtKing>

101	bamacsguy ⁹⁷²	
102	Blender Foundation	GPL
103	Blender Foundation	GPL
104	bamacsguy ⁹⁷³	
105	Blender Foundation	GPL
106	Blender Foundation	GPL
107	Blender Foundation	GPL
108	Blender Foundation	GPL
109	Blender Foundation	GPL
110	PvtKing ⁹⁷⁴	cc-by-sa-3.0
111	Blender Foundation	GPL
112		
113	Blender Foundation	GPL
114	Blender Foundation	GPL
115	Blender Foundation	GPL
116	JamesNZ ⁹⁷⁵	PD
117	JamesNZ ⁹⁷⁶	PD
118	JamesNZ ⁹⁷⁷	PD
119	JamesNZ ⁹⁷⁸	
120	Jake.lewis4	GPL
121	Jake.lewis4	GPL
122	JamesNZ ⁹⁷⁹	PD
123	Toadster ⁹⁸⁰	cc-by-sa-3.0
124	Myself	GPL
125	JamesNZ ⁹⁸¹	PD
126	Jake.lewis4	GPL
127	Jake.lewis4	GPL
128	JamesNZ ⁹⁸²	PD
129	JamesNZ ⁹⁸³	PD
130	JamesNZ ⁹⁸⁴	PD
131	JamesNZ ⁹⁸⁵	PD
132	Blender Foundation	GPL
133	Blender Foundation	GPL
134	Blender Foundation	GPL

⁹⁷² <http://en.wikibooks.org/wiki/User%3Abamacsguy>

⁹⁷³ <http://en.wikibooks.org/wiki/User%3Abamacsguy>

⁹⁷⁴ <http://en.wikibooks.org/wiki/User%3APvtKing>

⁹⁷⁵ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁷⁶ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁷⁷ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁷⁸ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁷⁹ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁸⁰ <http://en.wikibooks.org/wiki/User%3AToadster>

⁹⁸¹ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁸² <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁸³ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁸⁴ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

⁹⁸⁵ <http://en.wikibooks.org/wiki/User%3AJamesNZ>

135	Original uploader was Wwwwolf ⁹⁸⁶ at en.wikibooks ⁹⁸⁷ . Later version(s) were uploaded by Spiderworm ⁹⁸⁸ at en.wikibooks ⁹⁸⁹ .	
136	Original uploader was Wwwwolf ⁹⁹⁰ at en.wikibooks ⁹⁹¹	
137	Blender Foundation	GPL
138	Blender Foundation	GPL
139	Blender Foundation	GPL
140	Blender Foundation	GPL
141	Blender Foundation	GPL
142	Blender Foundation	GPL
143	Blender Foundation	GPL
144	Blender Foundation	GPL
145	Blender Foundation	GPL
146	Blender Foundation	GPL
147	Blender Foundation	GPL
148	Blender Foundation	GPL
149	Falstaff ⁹⁹²	GFDL
150	Blender Foundation	GPL
151	Blender Foundation	GPL
152	Blender Foundation ⁹⁹³	GPL
153	Blender Foundation ⁹⁹⁴	GPL
154	Blender Foundation ⁹⁹⁵	GPL
155	Ldo ⁹⁹⁶	cc-by-sa-3.0
156	Ldo ⁹⁹⁷	cc-by-sa-3.0
157	Ldo ⁹⁹⁸	cc-by-sa-3.0
158	Blender Foundation ⁹⁹⁹	GPL
159	Blender Foundation ¹⁰⁰⁰	GPL
160	Falstaff ¹⁰⁰¹	GFDL
161		
162	SoylentGreen ¹⁰⁰²	GPL
163	SoylentGreen ¹⁰⁰³	
164	Original uploader was Siesta ¹⁰⁰⁴ at en.wikibooks ¹⁰⁰⁵	GFDL

- 986 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWwwwolf>
- 987 <http://en.wikibooks.org>
- 988 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASpiderworm>
- 989 <http://en.wikibooks.org>
- 990 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWwwwolf>
- 991 <http://en.wikibooks.org>
- 992 <http://en.wikibooks.org/wiki/User%3AFalstaff>
- 993 <http://en.wikibooks.org/wiki/Blender%20Foundation>
- 994 <http://en.wikibooks.org/wiki/Blender%20Foundation>
- 995 <http://en.wikibooks.org/wiki/Blender%20Foundation>
- 996 <http://en.wikibooks.org/wiki/User%3ALdo>
- 997 <http://en.wikibooks.org/wiki/User%3ALdo>
- 998 <http://en.wikibooks.org/wiki/User%3ALdo>
- 999 <http://en.wikibooks.org/wiki/Blender%20Foundation>
- 1000 <http://en.wikibooks.org/wiki/Blender%20Foundation>
- 1001 <http://en.wikibooks.org/wiki/User%3AFalstaff>
- 1002 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
- 1003 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
- 1004 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASiesta>
- 1005 <http://en.wikibooks.org>

165	Weeix ¹⁰⁰⁶	
166	Original uploader was Nuclearpidgeon ¹⁰⁰⁷ at en.wikibooks ¹⁰⁰⁸	
167	Original uploader was Nuclearpidgeon ¹⁰⁰⁹ at en.wikibooks ¹⁰¹⁰	
168	Original uploader was Nuclearpidgeon ¹⁰¹¹ at en.wikibooks ¹⁰¹²	
169	Original uploader was Nuclearpidgeon ¹⁰¹³ at en.wikibooks ¹⁰¹⁴	
170	Amigosg ¹⁰¹⁵	cc-by-sa-3.0
171	Original uploader was Nuclearpidgeon ¹⁰¹⁶ at en.wikibooks ¹⁰¹⁷	
172	Amigosg ¹⁰¹⁸	PD
173	Original uploader was Nuclearpidgeon ¹⁰¹⁹ at en.wikibooks ¹⁰²⁰	
174	Original uploader was Nuclearpidgeon ¹⁰²¹ at en.wikibooks ¹⁰²²	
175	Original uploader was Nuclearpidgeon ¹⁰²³ at en.wikibooks ¹⁰²⁴	
176	Amigosg ¹⁰²⁵	PD
177	pearts ¹⁰²⁶	
178		cc-by-sa-2.5
179		cc-by-sa-2.5
180		cc-by-sa-2.5
181		cc-by-sa-2.5
182		cc-by-sa-2.5
183	pearts ¹⁰²⁷	
184		cc-by-sa-2.5
185	Original uploader was Megaloman ¹⁰²⁸ at en.wikibooks ¹⁰²⁹	
186	Original uploader was Lirtferk ¹⁰³⁰ at en.wikibooks ¹⁰³¹	GPL
187	Original uploader was Megaloman ¹⁰³² at en.wikibooks ¹⁰³³	

1006 <http://en.wikibooks.org/wiki/User%3AWeeix>

1007 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1008 <http://en.wikibooks.org>

1009 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1010 <http://en.wikibooks.org>

1011 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1012 <http://en.wikibooks.org>

1013 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1014 <http://en.wikibooks.org>

1015 <http://en.wikibooks.org/wiki/User%3AAmigosg>

1016 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1017 <http://en.wikibooks.org>

1018 <http://en.wikibooks.org/wiki/User%3AAmigosg>

1019 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1020 <http://en.wikibooks.org>

1021 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1022 <http://en.wikibooks.org>

1023 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

1024 <http://en.wikibooks.org>

1025 <http://en.wikibooks.org/wiki/User%3AAmigosg>

1026 <http://en.wikibooks.org/wiki/User%3Apearts>

1027 <http://en.wikibooks.org/wiki/User%3Apearts>

1028 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>

1029 <http://en.wikibooks.org>

1030 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALirtferk>

1031 <http://en.wikibooks.org>

1032 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>

1033 <http://en.wikibooks.org>

188	Original uploader was Megaloman ¹⁰³⁴ at en.wikibooks ¹⁰³⁵	GPL
189	Original uploader was Megaloman ¹⁰³⁶ at en.wikibooks ¹⁰³⁷	GPL
190	Hendric ¹⁰³⁸	GFDL
191	Hendric ¹⁰³⁹	GFDL
192	Original uploader was Megaloman ¹⁰⁴⁰ at en.wikibooks ¹⁰⁴¹	GPL
193	Original uploader was Skeep ¹⁰⁴² at en.wikibooks ¹⁰⁴³	GPL
194	Koltonaugust ¹⁰⁴⁴	cc-by-sa-3.0
195	Original uploader was Morrolan ¹⁰⁴⁵ at en.wikibooks ¹⁰⁴⁶ . Later version(s) were uploaded by Adrignola ¹⁰⁴⁷ at en.wikibooks ¹⁰⁴⁸ .	GPL
196	Original uploader was Morrolan ¹⁰⁴⁹ at en.wikibooks ¹⁰⁵⁰ . Later version(s) were uploaded by Adrignola ¹⁰⁵¹ at en.wikibooks ¹⁰⁵² .	GPL
197	Original uploader was Morrolan ¹⁰⁵³ at en.wikibooks ¹⁰⁵⁴ . Later version(s) were uploaded by Adrignola ¹⁰⁵⁵ at en.wikibooks ¹⁰⁵⁶ .	GPL
198	Original uploader was Morrolan ¹⁰⁵⁷ at en.wikibooks ¹⁰⁵⁸ . Later version(s) were uploaded by Adrignola ¹⁰⁵⁹ at en.wikibooks ¹⁰⁶⁰ .	GPL
199	Original uploader was Morrolan ¹⁰⁶¹ at en.wikibooks ¹⁰⁶² . Later version(s) were uploaded by Adrignola ¹⁰⁶³ at en.wikibooks ¹⁰⁶⁴ .	GPL
200	Original uploader was Morrolan ¹⁰⁶⁵ at en.wikibooks ¹⁰⁶⁶ . Later version(s) were uploaded by Adrignola ¹⁰⁶⁷ at en.wikibooks ¹⁰⁶⁸ .	GPL

1034 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>

1035 <http://en.wikibooks.org>

1036 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>

1037 <http://en.wikibooks.org>

1038 <http://en.wikibooks.org/wiki/User%3AHendric>

1039 <http://en.wikibooks.org/wiki/User%3AHendric>

1040 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>

1041 <http://en.wikibooks.org>

1042 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASkeep>

1043 <http://en.wikibooks.org>

1044 <http://en.wikibooks.org/wiki/User%3AKoltonaugust>

1045 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1046 <http://en.wikibooks.org>

1047 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1048 <http://en.wikibooks.org>

1049 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1050 <http://en.wikibooks.org>

1051 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1052 <http://en.wikibooks.org>

1053 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1054 <http://en.wikibooks.org>

1055 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1056 <http://en.wikibooks.org>

1057 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1058 <http://en.wikibooks.org>

1059 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1060 <http://en.wikibooks.org>

1061 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1062 <http://en.wikibooks.org>

1063 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1064 <http://en.wikibooks.org>

1065 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1066 <http://en.wikibooks.org>

1067 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1068 <http://en.wikibooks.org>

201	Original uploader was Morrolan ¹⁰⁶⁹ at en.wikibooks ¹⁰⁷⁰ . Later version(s) were uploaded by Adrignola ¹⁰⁷¹ at en.wikibooks ¹⁰⁷² .	GPL
202	Original uploader was Morrolan ¹⁰⁷³ at en.wikibooks ¹⁰⁷⁴ . Later version(s) were uploaded by Adrignola ¹⁰⁷⁵ at en.wikibooks ¹⁰⁷⁶ .	GPL
203	Original uploader was Morrolan ¹⁰⁷⁷ at en.wikibooks ¹⁰⁷⁸	GPL
204	Original uploader was Morrolan ¹⁰⁷⁹ at en.wikibooks ¹⁰⁸⁰	GPL
205	Original uploader was Morrolan ¹⁰⁸¹ at en.wikibooks ¹⁰⁸² . Later version(s) were uploaded by Adrignola ¹⁰⁸³ at en.wikibooks ¹⁰⁸⁴ .	GPL
206	Original uploader was Morrolan ¹⁰⁸⁵ at en.wikibooks ¹⁰⁸⁶	GPL
207	Original uploader was Morrolan ¹⁰⁸⁷ at en.wikibooks ¹⁰⁸⁸	GPL
208	Blender Foundation	GPL
209	Original uploader was Morrolan ¹⁰⁸⁹ at en.wikibooks ¹⁰⁹⁰	GPL
210	Original uploader was Morrolan ¹⁰⁹¹ at en.wikibooks ¹⁰⁹²	GPL
211		
212	Original uploader was Shythinker ¹⁰⁹³ at en.wikibooks ¹⁰⁹⁴	GFDL
213	Romain Behar. Original uploader was Romainbehar ¹⁰⁹⁵ at en.wikibooks ¹⁰⁹⁶	PD
214		
215	Screenshot of Blender by me.	GPL
216	Original uploader was Shythinker ¹⁰⁹⁷ at en.wikibooks ¹⁰⁹⁸	GFDL
217	Original uploader was Shythinker ¹⁰⁹⁹ at en.wikibooks ¹¹⁰⁰	GFDL

1069 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1070 <http://en.wikibooks.org>

1071 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1072 <http://en.wikibooks.org>

1073 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1074 <http://en.wikibooks.org>

1075 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1076 <http://en.wikibooks.org>

1077 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1078 <http://en.wikibooks.org>

1079 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1080 <http://en.wikibooks.org>

1081 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1082 <http://en.wikibooks.org>

1083 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

1084 <http://en.wikibooks.org>

1085 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1086 <http://en.wikibooks.org>

1087 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1088 <http://en.wikibooks.org>

1089 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1090 <http://en.wikibooks.org>

1091 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMorrolan>

1092 <http://en.wikibooks.org>

1093 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1094 <http://en.wikibooks.org>

1095 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1096 <http://en.wikibooks.org>

1097 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1098 <http://en.wikibooks.org>

1099 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1100 <http://en.wikibooks.org>

218	Original uploader was Shythinker ¹¹⁰¹ at en.wikibooks ¹¹⁰²	GFDL
219		GPL
220	Original uploader was Shythinker ¹¹⁰³ at en.wikibooks ¹¹⁰⁴	GFDL
221	Romain Behar. Original uploader was Romainbehar ¹¹⁰⁵ at en.wikibooks ¹¹⁰⁶	PD
222	Original uploader was Shythinker ¹¹⁰⁷ at en.wikibooks ¹¹⁰⁸	GFDL
223	Boyage ¹¹⁰⁹ . Original uploader was Boyage ¹¹¹⁰ at en.wikibooks ¹¹¹¹	GFDL
224	Boyage ¹¹¹² . Original uploader was Boyage ¹¹¹³ at en.wikibooks ¹¹¹⁴	GFDL
225	Boyage ¹¹¹⁵ . Original uploader was Boyage ¹¹¹⁶ at en.wikibooks ¹¹¹⁷	GFDL
226	Romain Behar. Original uploader was Romainbehar ¹¹¹⁸ at en.wikibooks ¹¹¹⁹	PD
227	Original uploader was Shythinker ¹¹²⁰ at en.wikibooks ¹¹²¹	GFDL
228	Romain Behar. Original uploader was Romainbehar ¹¹²² at en.wikibooks ¹¹²³	PD
229	Boyage ¹¹²⁴ . Original uploader was Boyage ¹¹²⁵ at en.wikibooks ¹¹²⁶	GFDL
230	Original uploader was Shythinker ¹¹²⁷ at en.wikibooks ¹¹²⁸	GFDL
231	Romain Behar. Original uploader was Romainbehar ¹¹²⁹ at en.wikibooks ¹¹³⁰	PD

1101 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1102 <http://en.wikibooks.org>

1103 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1104 <http://en.wikibooks.org>

1105 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1106 <http://en.wikibooks.org>

1107 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1108 <http://en.wikibooks.org>

1109 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1110 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1111 <http://en.wikibooks.org>

1112 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1113 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1114 <http://en.wikibooks.org>

1115 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1116 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1117 <http://en.wikibooks.org>

1118 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1119 <http://en.wikibooks.org>

1120 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1121 <http://en.wikibooks.org>

1122 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1123 <http://en.wikibooks.org>

1124 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1125 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1126 <http://en.wikibooks.org>

1127 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1128 <http://en.wikibooks.org>

1129 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1130 <http://en.wikibooks.org>

232	Romain Behar. en.wikibooks ¹¹³²	Original uploader was Romainbehar ¹¹³¹ at	PD
233	Boyage ¹¹³³ . en.wikibooks ¹¹³⁵	Original uploader was Boyage ¹¹³⁴ at	GFDL
234	Boyage ¹¹³⁶ . en.wikibooks ¹¹³⁸	Original uploader was Boyage ¹¹³⁷ at	GFDL
235	Romain Behar. en.wikibooks ¹¹⁴⁰	Original uploader was Romainbehar ¹¹³⁹ at	PD
236	Boyage ¹¹⁴¹ . en.wikibooks ¹¹⁴³	Original uploader was Boyage ¹¹⁴² at	GFDL
237	Original uploader was Shythinker ¹¹⁴⁴ at en.wikibooks ¹¹⁴⁵		GFDL
238	Boyage ¹¹⁴⁶ . en.wikibooks ¹¹⁴⁸	Original uploader was Boyage ¹¹⁴⁷ at	GFDL
239	Boyage ¹¹⁴⁹ . en.wikibooks ¹¹⁵¹	Original uploader was Boyage ¹¹⁵⁰ at	GFDL
240	Romain Behar. en.wikibooks ¹¹⁵³	Original uploader was Romainbehar ¹¹⁵² at	PD
241	Romain Behar. en.wikibooks ¹¹⁵⁵	Original uploader was Romainbehar ¹¹⁵⁴ at	PD
242			PD
243	Jesse S. (Bnty)		PD
244			PD
245	Original uploader was Wynnmc ¹¹⁵⁶ at en.wikibooks ¹¹⁵⁷		
246			PD
247			PD

1131 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1132 <http://en.wikibooks.org>

1133 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1134 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1135 <http://en.wikibooks.org>

1136 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1137 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1138 <http://en.wikibooks.org>

1139 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1140 <http://en.wikibooks.org>

1141 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1142 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1143 <http://en.wikibooks.org>

1144 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

1145 <http://en.wikibooks.org>

1146 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1147 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1148 <http://en.wikibooks.org>

1149 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1150 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

1151 <http://en.wikibooks.org>

1152 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1153 <http://en.wikibooks.org>

1154 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1155 <http://en.wikibooks.org>

1156 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1157 <http://en.wikibooks.org>

248	Romain Behar. Original uploader was Romainbehar ¹¹⁵⁸ at en.wikibooks ¹¹⁵⁹	PD
249	Original uploader was Wynnmc ¹¹⁶⁰ at en.wikibooks ¹¹⁶¹	
250		GFDL
251	Original uploader was Wynnmc ¹¹⁶² at en.wikibooks ¹¹⁶³	
252		GFDL
253		GFDL
254	Original uploader was Wynnmc ¹¹⁶⁴ at en.wikibooks ¹¹⁶⁵	
255	Original uploader was Wynnmc ¹¹⁶⁶ at en.wikibooks ¹¹⁶⁷	
256	Original uploader was Wynnmc ¹¹⁶⁸ at en.wikibooks ¹¹⁶⁹	
257	Original uploader was Wynnmc ¹¹⁷⁰ at en.wikibooks ¹¹⁷¹	
258	Original uploader was Wynnmc ¹¹⁷² at en.wikibooks ¹¹⁷³	
259	Original uploader was Wynnmc ¹¹⁷⁴ at en.wikibooks ¹¹⁷⁵	
260	Original uploader was Wynnmc ¹¹⁷⁶ at en.wikibooks ¹¹⁷⁷	
261	Original uploader was Wynnmc ¹¹⁷⁸ at en.wikibooks ¹¹⁷⁹	
262	Original uploader was Wynnmc ¹¹⁸⁰ at en.wikibooks ¹¹⁸¹	
263	Original uploader was Wynnmc ¹¹⁸² at en.wikibooks ¹¹⁸³	
264	Original uploader was Wynnmc ¹¹⁸⁴ at en.wikibooks ¹¹⁸⁵	
265	Original uploader was Wynnmc ¹¹⁸⁶ at en.wikibooks ¹¹⁸⁷	
266	Original uploader was Wynnmc ¹¹⁸⁸ at en.wikibooks ¹¹⁸⁹	

1158 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

1159 <http://en.wikibooks.org>

1160 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1161 <http://en.wikibooks.org>

1162 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1163 <http://en.wikibooks.org>

1164 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1165 <http://en.wikibooks.org>

1166 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1167 <http://en.wikibooks.org>

1168 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1169 <http://en.wikibooks.org>

1170 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1171 <http://en.wikibooks.org>

1172 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1173 <http://en.wikibooks.org>

1174 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1175 <http://en.wikibooks.org>

1176 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1177 <http://en.wikibooks.org>

1178 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1179 <http://en.wikibooks.org>

1180 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1181 <http://en.wikibooks.org>

1182 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1183 <http://en.wikibooks.org>

1184 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1185 <http://en.wikibooks.org>

1186 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1187 <http://en.wikibooks.org>

1188 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

1189 <http://en.wikibooks.org>

267	Original uploader was Wynnmc ¹¹⁹⁰ at en.wikibooks ¹¹⁹¹	
268	Original uploader was Wynnmc ¹¹⁹² at en.wikibooks ¹¹⁹³	
269	Original uploader was Wynnmc ¹¹⁹⁴ at en.wikibooks ¹¹⁹⁵	
270	Original uploader was Wynnmc ¹¹⁹⁶ at en.wikibooks ¹¹⁹⁷	
271	Original uploader was Wynnmc ¹¹⁹⁸ at en.wikibooks ¹¹⁹⁹	
272	Original uploader was Wynnmc ¹²⁰⁰ at en.wikibooks ¹²⁰¹	
273	Original uploader was Wynnmc ¹²⁰² at en.wikibooks ¹²⁰³	
274	Original uploader was Wynnmc ¹²⁰⁴ at en.wikibooks ¹²⁰⁵	
275	Original uploader was Wynnmc ¹²⁰⁶ at en.wikibooks ¹²⁰⁷	
276	Original uploader was Wynnmc ¹²⁰⁸ at en.wikibooks ¹²⁰⁹	
277	Original uploader was Wynnmc ¹²¹⁰ at en.wikibooks ¹²¹¹	
278	Original uploader was Wynnmc ¹²¹² at en.wikibooks ¹²¹³	
279	Original uploader was Wynnmc ¹²¹⁴ at en.wikibooks ¹²¹⁵	
280	Original uploader was Wynnmc ¹²¹⁶ at en.wikibooks ¹²¹⁷	
281		GFDL
282	Original uploader was Wynnmc ¹²¹⁸ at en.wikibooks ¹²¹⁹	
283	Original uploader was Wynnmc ¹²²⁰ at en.wikibooks ¹²²¹	
284	Original uploader was Wynnmc ¹²²² at en.wikibooks ¹²²³	

¹¹⁹⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹¹⁹¹ <http://en.wikibooks.org>

¹¹⁹² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹¹⁹³ <http://en.wikibooks.org>

¹¹⁹⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹¹⁹⁵ <http://en.wikibooks.org>

¹¹⁹⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹¹⁹⁷ <http://en.wikibooks.org>

¹¹⁹⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹¹⁹⁹ <http://en.wikibooks.org>

¹²⁰⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²⁰¹ <http://en.wikibooks.org>

¹²⁰² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²⁰³ <http://en.wikibooks.org>

¹²⁰⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²⁰⁵ <http://en.wikibooks.org>

¹²⁰⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²⁰⁷ <http://en.wikibooks.org>

¹²⁰⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²⁰⁹ <http://en.wikibooks.org>

¹²¹⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²¹¹ <http://en.wikibooks.org>

¹²¹² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²¹³ <http://en.wikibooks.org>

¹²¹⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²¹⁵ <http://en.wikibooks.org>

¹²¹⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²¹⁷ <http://en.wikibooks.org>

¹²¹⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²¹⁹ <http://en.wikibooks.org>

¹²²⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²²¹ <http://en.wikibooks.org>

¹²²² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²²³ <http://en.wikibooks.org>

285	Original uploader was Wynnmc ¹²²⁴ at en.wikibooks ¹²²⁵	
286	<ul style="list-style-type: none"> • Blender: w:Blender Foundation¹²²⁶ w:¹²²⁷ 	GPL
287	Original uploader was Shythinker ¹²²⁸ at en.wikibooks ¹²²⁹	GFDL
288	Original uploader was Shythinker ¹²³⁰ at en.wikibooks ¹²³¹	GFDL
289		PD
290		PD
291	Boyage ¹²³² . Original uploader was Boyage ¹²³³ at en.wikibooks ¹²³⁴	GFDL
292	Original uploader was Shythinker ¹²³⁵ at en.wikibooks ¹²³⁶	GFDL
293	Original uploader was Shythinker ¹²³⁷ at en.wikibooks ¹²³⁸	GFDL
294	Boyage ¹²³⁹ . Original uploader was Boyage ¹²⁴⁰ at en.wikibooks ¹²⁴¹	
295	Original uploader was Shythinker ¹²⁴² at en.wikibooks ¹²⁴³	GFDL
0	Loremus Ipsumus	None
0	Loremus Ipsumus	None
298	Original uploader was Shythinker ¹²⁴⁴ at en.wikibooks ¹²⁴⁵	GFDL
299	Original uploader was Shythinker ¹²⁴⁶ at en.wikibooks ¹²⁴⁷	GFDL
300	Original uploader was Shythinker ¹²⁴⁸ at en.wikibooks ¹²⁴⁹	GFDL
0	Loremus Ipsumus	None
302	Original uploader was Shythinker ¹²⁵⁰ at en.wikibooks ¹²⁵¹	GFDL

¹²²⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹²²⁵ <http://en.wikibooks.org>

¹²²⁶ <http://en.wikipedia.org/wiki/Blender%20Foundation>

¹²²⁷ <http://en.wikipedia.org/wiki/%0A>

¹²²⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²²⁹ <http://en.wikibooks.org>

¹²³⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²³¹ <http://en.wikibooks.org>

¹²³² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

¹²³³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

¹²³⁴ <http://en.wikibooks.org>

¹²³⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²³⁶ <http://en.wikibooks.org>

¹²³⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²³⁸ <http://en.wikibooks.org>

¹²³⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

¹²⁴⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>

¹²⁴¹ <http://en.wikibooks.org>

¹²⁴² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²⁴³ <http://en.wikibooks.org>

¹²⁴⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²⁴⁵ <http://en.wikibooks.org>

¹²⁴⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²⁴⁷ <http://en.wikibooks.org>

¹²⁴⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²⁴⁹ <http://en.wikibooks.org>

¹²⁵⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>

¹²⁵¹ <http://en.wikibooks.org>

303	Boyage ¹²⁵² . en.wikibooks ¹²⁵⁴	Original uploader was Boyage ¹²⁵³ at	
304	Boyage ¹²⁵⁵ . en.wikibooks ¹²⁵⁷	Original uploader was Boyage ¹²⁵⁶ at	
305	Boyage ¹²⁵⁸ . en.wikibooks ¹²⁶⁰	Original uploader was Boyage ¹²⁵⁹ at	
306			PD
307			PD
308	Boyage ¹²⁶¹ . en.wikibooks ¹²⁶³	Original uploader was Boyage ¹²⁶² at	
309	Boyage ¹²⁶⁴ . en.wikibooks ¹²⁶⁶	Original uploader was Boyage ¹²⁶⁵ at	
310	Boyage ¹²⁶⁷ . en.wikibooks ¹²⁶⁹	Original uploader was Boyage ¹²⁶⁸ at	
311	Boyage ¹²⁷⁰ . en.wikibooks ¹²⁷²	Original uploader was Boyage ¹²⁷¹ at	
312	Boyage ¹²⁷³ . en.wikibooks ¹²⁷⁵	Original uploader was Boyage ¹²⁷⁴ at	GFDL
313	Original uploader was Shythinker ¹²⁷⁶ at en.wikibooks ¹²⁷⁷		GFDL
314	Boyage ¹²⁷⁸ . en.wikibooks ¹²⁸⁰	Original uploader was Boyage ¹²⁷⁹ at	

1252 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1253 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1254 <http://en.wikibooks.org>
1255 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1256 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1257 <http://en.wikibooks.org>
1258 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1259 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1260 <http://en.wikibooks.org>
1261 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1262 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1263 <http://en.wikibooks.org>
1264 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1265 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1266 <http://en.wikibooks.org>
1267 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1268 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1269 <http://en.wikibooks.org>
1270 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1271 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1272 <http://en.wikibooks.org>
1273 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1274 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1275 <http://en.wikibooks.org>
1276 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>
1277 <http://en.wikibooks.org>
1278 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1279 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1280 <http://en.wikibooks.org>

315	Boyage ¹²⁸¹ . en.wikibooks ¹²⁸³	Original uploader was Boyage ¹²⁸² at	
316	Boyage ¹²⁸⁴ . en.wikibooks ¹²⁸⁶	Original uploader was Boyage ¹²⁸⁵ at	
317	Original uploader was Shythinker ¹²⁸⁷ at en.wikibooks ¹²⁸⁸		GFDL
318	Arch dude ¹²⁸⁹		GFDL
319	Arch dude ¹²⁹⁰		GFDL
320	pearts ¹²⁹¹		
321	pearts ¹²⁹²		
322	Original uploader was AverieZen ¹²⁹³ at en.wikibooks ¹²⁹⁴		
323	Original uploader was AverieZen ¹²⁹⁵ at en.wikibooks ¹²⁹⁶		
324	Original uploader was AverieZen ¹²⁹⁷ at en.wikibooks ¹²⁹⁸		
325	Original uploader was AverieZen ¹²⁹⁹ at en.wikibooks ¹³⁰⁰		GPL
326	Original uploader was Eyeonus ¹³⁰¹ at en.wikibooks ¹³⁰²		
327	Original uploader was Eyeonus ¹³⁰³ at en.wikibooks ¹³⁰⁴		
328			PD
329	Original uploader was Wynnmc ¹³⁰⁵ at en.wikibooks ¹³⁰⁶		
330	Original uploader was Wynnmc ¹³⁰⁷ at en.wikibooks ¹³⁰⁸		
331	Original uploader was Wynnmc ¹³⁰⁹ at en.wikibooks ¹³¹⁰		
332	Original uploader was Wynnmc ¹³¹¹ at en.wikibooks ¹³¹²		

1281 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1282 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1283 <http://en.wikibooks.org>
1284 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1285 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
1286 <http://en.wikibooks.org>
1287 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AShythinker>
1288 <http://en.wikibooks.org>
1289 <http://en.wikibooks.org/wiki/User%3AArch%20dude>
1290 <http://en.wikibooks.org/wiki/User%3AArch%20dude>
1291 <http://en.wikibooks.org/wiki/User%3Apearts>
1292 <http://en.wikibooks.org/wiki/User%3Apearts>
1293 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1294 <http://en.wikibooks.org>
1295 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1296 <http://en.wikibooks.org>
1297 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1298 <http://en.wikibooks.org>
1299 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1300 <http://en.wikibooks.org>
1301 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AEyeonus>
1302 <http://en.wikibooks.org>
1303 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AEyeonus>
1304 <http://en.wikibooks.org>
1305 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1306 <http://en.wikibooks.org>
1307 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1308 <http://en.wikibooks.org>
1309 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1310 <http://en.wikibooks.org>
1311 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1312 <http://en.wikibooks.org>

333	Original uploader was Wynnmc ¹³¹³ at en.wikibooks ¹³¹⁴	
334	André Karwath aka Aka ¹³¹⁵	cc-by-sa-2.5
335	Arch dude ¹³¹⁶	GFDL
336	Arch dude ¹³¹⁷	GFDL
337	Original uploader was AverieZen ¹³¹⁸ at en.wikibooks ¹³¹⁹	
338	Original uploader was AverieZen ¹³²⁰ at en.wikibooks ¹³²¹	
339	Original uploader was AverieZen ¹³²² at en.wikibooks ¹³²³	
340	Original uploader was AverieZen ¹³²⁴ at en.wikibooks ¹³²⁵	
341	Original uploader was AverieZen ¹³²⁶ at en.wikibooks ¹³²⁷	
342	Original uploader was AverieZen ¹³²⁸ at en.wikibooks ¹³²⁹	
343	Original uploader was AverieZen ¹³³⁰ at en.wikibooks ¹³³¹	
344	Original uploader was AverieZen ¹³³² at en.wikibooks ¹³³³	
345	Original uploader was AverieZen ¹³³⁴ at en.wikibooks ¹³³⁵	
346	Original uploader was Clone Dad ¹³³⁶ at en.wikibooks ¹³³⁷	GFDL
347	Original uploader was AverieZen ¹³³⁸ at en.wikibooks ¹³³⁹	
348	Slayerjp ¹³⁴⁰	cc-by-sa-3.0
349	Original uploader was AverieZen ¹³⁴¹ at en.wikibooks ¹³⁴²	
350	Original uploader was AverieZen ¹³⁴³ at en.wikibooks ¹³⁴⁴	
351	Original uploader was AverieZen ¹³⁴⁵ at en.wikibooks ¹³⁴⁶	

¹³¹³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>

¹³¹⁴ <http://en.wikibooks.org>

¹³¹⁵ <http://en.wikibooks.org/wiki/user%3AAka>

¹³¹⁶ <http://en.wikibooks.org/wiki/User%3AArch%20dude>

¹³¹⁷ <http://en.wikibooks.org/wiki/User%3AArch%20dude>

¹³¹⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³¹⁹ <http://en.wikibooks.org>

¹³²⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³²¹ <http://en.wikibooks.org>

¹³²² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³²³ <http://en.wikibooks.org>

¹³²⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³²⁵ <http://en.wikibooks.org>

¹³²⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³²⁷ <http://en.wikibooks.org>

¹³²⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³²⁹ <http://en.wikibooks.org>

¹³³⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³³¹ <http://en.wikibooks.org>

¹³³² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³³³ <http://en.wikibooks.org>

¹³³⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³³⁵ <http://en.wikibooks.org>

¹³³⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AClone%20Dad>

¹³³⁷ <http://en.wikibooks.org>

¹³³⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³³⁹ <http://en.wikibooks.org>

¹³⁴⁰ <http://en.wikibooks.org/wiki/User%3ASlayerjp>

¹³⁴¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁴² <http://en.wikibooks.org>

¹³⁴³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁴⁴ <http://en.wikibooks.org>

¹³⁴⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁴⁶ <http://en.wikibooks.org>

352	Original uploader was AverieZen ¹³⁴⁷ at en.wikibooks ¹³⁴⁸	
353	Original uploader was AverieZen ¹³⁴⁹ at en.wikibooks ¹³⁵⁰	
354	Original uploader was AverieZen ¹³⁵¹ at en.wikibooks ¹³⁵²	
355	Original uploader was AverieZen ¹³⁵³ at en.wikibooks ¹³⁵⁴	
356	Original uploader was AverieZen ¹³⁵⁵ at en.wikibooks ¹³⁵⁶	
357	Original uploader was AverieZen ¹³⁵⁷ at en.wikibooks ¹³⁵⁸	
358	Original uploader was AverieZen ¹³⁵⁹ at en.wikibooks ¹³⁶⁰	
359	Original uploader was AverieZen ¹³⁶¹ at en.wikibooks ¹³⁶²	
360	Original uploader was AverieZen ¹³⁶³ at en.wikibooks ¹³⁶⁴	
361	Original uploader was Josellis ¹³⁶⁵ at en.wikibooks ¹³⁶⁶	
362	Original uploader was AverieZen ¹³⁶⁷ at en.wikibooks ¹³⁶⁸	
363	Original uploader was AverieZen ¹³⁶⁹ at en.wikibooks ¹³⁷⁰	
364	Original uploader was AverieZen ¹³⁷¹ at en.wikibooks ¹³⁷²	
365	Original uploader was AverieZen ¹³⁷³ at en.wikibooks ¹³⁷⁴	
366	Original uploader was AverieZen ¹³⁷⁵ at en.wikibooks ¹³⁷⁶	
367	pearts ¹³⁷⁷	
368		PD
369	Original uploader was AverieZen ¹³⁷⁸ at en.wikibooks ¹³⁷⁹	

¹³⁴⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁴⁸ <http://en.wikibooks.org>

¹³⁴⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁵⁰ <http://en.wikibooks.org>

¹³⁵¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁵² <http://en.wikibooks.org>

¹³⁵³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁵⁴ <http://en.wikibooks.org>

¹³⁵⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁵⁶ <http://en.wikibooks.org>

¹³⁵⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁵⁸ <http://en.wikibooks.org>

¹³⁵⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁶⁰ <http://en.wikibooks.org>

¹³⁶¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁶² <http://en.wikibooks.org>

¹³⁶³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁶⁴ <http://en.wikibooks.org>

¹³⁶⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJosellis>

¹³⁶⁶ <http://en.wikibooks.org>

¹³⁶⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁶⁸ <http://en.wikibooks.org>

¹³⁶⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁷⁰ <http://en.wikibooks.org>

¹³⁷¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁷² <http://en.wikibooks.org>

¹³⁷³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁷⁴ <http://en.wikibooks.org>

¹³⁷⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁷⁶ <http://en.wikibooks.org>

¹³⁷⁷ <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁷⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁷⁹ <http://en.wikibooks.org>

370	Original uploader was AverieZen ¹³⁸⁰ at en.wikibooks ¹³⁸¹	
371	pearts ¹³⁸²	
372	Original uploader was AverieZen ¹³⁸³ at en.wikibooks ¹³⁸⁴	
373		GFDL
374	Original uploader was AverieZen ¹³⁸⁵ at en.wikibooks ¹³⁸⁶	
375	pearts ¹³⁸⁷	
376	Original uploader was AverieZen ¹³⁸⁸ at en.wikibooks ¹³⁸⁹	
377	Original uploader was AverieZen ¹³⁹⁰ at en.wikibooks ¹³⁹¹	
378	Original uploader was AverieZen ¹³⁹² at en.wikibooks ¹³⁹³	
379	pearts ¹³⁹⁴	
380	pearts ¹³⁹⁵	
381	pearts ¹³⁹⁶	
382	Original uploaded by AverieZen ¹³⁹⁷ (Transferred by pearts ¹³⁹⁸)	
383	pearts ¹³⁹⁹	
384	Original uploader was AverieZen ¹⁴⁰⁰ at en.wikibooks ¹⁴⁰¹	
385	pearts ¹⁴⁰²	
386	pearts ¹⁴⁰³	
387	pearts ¹⁴⁰⁴	
388	pearts ¹⁴⁰⁵	
389	pearts ¹⁴⁰⁶	
390	pearts ¹⁴⁰⁷	
391	pearts ¹⁴⁰⁸	
392	Paul James Chatman	PD

¹³⁸⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁸¹ <http://en.wikibooks.org>

¹³⁸² <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁸³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁸⁴ <http://en.wikibooks.org>

¹³⁸⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁸⁶ <http://en.wikibooks.org>

¹³⁸⁷ <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁸⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁸⁹ <http://en.wikibooks.org>

¹³⁹⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁹¹ <http://en.wikibooks.org>

¹³⁹² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁹³ <http://en.wikibooks.org>

¹³⁹⁴ <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁹⁵ <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁹⁶ <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁹⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹³⁹⁸ <http://en.wikibooks.org/wiki/User%3Apearts>

¹³⁹⁹ <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴⁰¹ <http://en.wikibooks.org>

¹⁴⁰² <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰³ <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰⁴ <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰⁵ <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰⁶ <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰⁷ <http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴⁰⁸ <http://en.wikibooks.org/wiki/User%3Apearts>

393	Paul James Chatman	PD
394	Paul James Chatman	PD
395	pearts ¹⁴⁰⁹	
396	pearts ¹⁴¹⁰	
397	pearts ¹⁴¹¹	
398	pearts ¹⁴¹²	
399	pearts ¹⁴¹³	
400	pearts ¹⁴¹⁴	
401	pearts ¹⁴¹⁵	
402	pearts ¹⁴¹⁶	
403	pearts ¹⁴¹⁷	
404	Original uploader was AverieZen ¹⁴¹⁸ at en.wikibooks ¹⁴¹⁹	
405	Original uploader was AverieZen ¹⁴²⁰ at en.wikibooks ¹⁴²¹	
406	pearts ¹⁴²²	
407	Original uploader was AverieZen ¹⁴²³ at en.wikibooks ¹⁴²⁴	
408	pearts ¹⁴²⁵	
409	pearts ¹⁴²⁶	
410	pearts ¹⁴²⁷	
411	pearts ¹⁴²⁸	
412	Original uploader was AverieZen ¹⁴²⁹ at en.wikibooks ¹⁴³⁰	
413	Original uploader was AverieZen ¹⁴³¹ at en.wikibooks ¹⁴³²	
414	Original uploader was AverieZen ¹⁴³³ at en.wikibooks ¹⁴³⁴	
415	Original uploader was AverieZen ¹⁴³⁵ at en.wikibooks ¹⁴³⁶	

¹⁴⁰⁹<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹⁰<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹¹<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹²<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹³<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹⁴<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹⁵<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹⁶<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹⁷<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴¹⁸<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴¹⁹<http://en.wikibooks.org>

¹⁴²⁰<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴²¹<http://en.wikibooks.org>

¹⁴²²<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴²³<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴²⁴<http://en.wikibooks.org>

¹⁴²⁵<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴²⁶<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴²⁷<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴²⁸<http://en.wikibooks.org/wiki/User%3Apearts>

¹⁴²⁹<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴³⁰<http://en.wikibooks.org>

¹⁴³¹<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴³²<http://en.wikibooks.org>

¹⁴³³<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴³⁴<http://en.wikibooks.org>

¹⁴³⁵<http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

¹⁴³⁶<http://en.wikibooks.org>

416	Original uploader was AverieZen ¹⁴³⁷ at en.wikibooks ¹⁴³⁸	
417	Original uploader was AverieZen ¹⁴³⁹ at en.wikibooks ¹⁴⁴⁰	
418	Original uploader was AverieZen ¹⁴⁴¹ at en.wikibooks ¹⁴⁴²	
419	Original uploader was AverieZen ¹⁴⁴³ at en.wikibooks ¹⁴⁴⁴	
420	Original uploader was Alexwill84 ¹⁴⁴⁵ at en.wikibooks ¹⁴⁴⁶	
421	Original uploader was Alexwill84 ¹⁴⁴⁷ at en.wikibooks ¹⁴⁴⁸	
422	Original uploader was Alexwill84 ¹⁴⁴⁹ at en.wikibooks ¹⁴⁵⁰	
423	Original uploader was Alexwill84 ¹⁴⁵¹ at en.wikibooks ¹⁴⁵²	
424	Original uploader was Alexwill84 ¹⁴⁵³ at en.wikibooks ¹⁴⁵⁴	
425	Original uploader was Alexwill84 ¹⁴⁵⁵ at en.wikibooks ¹⁴⁵⁶	
426	Original uploader was Alexwill84 ¹⁴⁵⁷ at en.wikibooks ¹⁴⁵⁸	
427	Original uploader was Alexwill84 ¹⁴⁵⁹ at en.wikibooks ¹⁴⁶⁰	
428	Original uploader was AverieZen ¹⁴⁶¹ at en.wikibooks ¹⁴⁶²	
429	Original uploader was AverieZen ¹⁴⁶³ at en.wikibooks ¹⁴⁶⁴	
430	Original uploader was AverieZen ¹⁴⁶⁵ at en.wikibooks ¹⁴⁶⁶	
431	Original uploader was AverieZen ¹⁴⁶⁷ at en.wikibooks ¹⁴⁶⁸	
432	Original uploader was AverieZen ¹⁴⁶⁹ at en.wikibooks ¹⁴⁷⁰	

-
- 1437 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1438 <http://en.wikibooks.org>
 - 1439 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1440 <http://en.wikibooks.org>
 - 1441 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1442 <http://en.wikibooks.org>
 - 1443 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1444 <http://en.wikibooks.org>
 - 1445 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1446 <http://en.wikibooks.org>
 - 1447 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1448 <http://en.wikibooks.org>
 - 1449 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1450 <http://en.wikibooks.org>
 - 1451 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1452 <http://en.wikibooks.org>
 - 1453 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1454 <http://en.wikibooks.org>
 - 1455 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1456 <http://en.wikibooks.org>
 - 1457 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1458 <http://en.wikibooks.org>
 - 1459 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAlexwill84>
 - 1460 <http://en.wikibooks.org>
 - 1461 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1462 <http://en.wikibooks.org>
 - 1463 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1464 <http://en.wikibooks.org>
 - 1465 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1466 <http://en.wikibooks.org>
 - 1467 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1468 <http://en.wikibooks.org>
 - 1469 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
 - 1470 <http://en.wikibooks.org>

433	Original uploader was AverieZen ¹⁴⁷¹ at en.wikibooks ¹⁴⁷²	
434	Original uploader was AverieZen ¹⁴⁷³ at en.wikibooks ¹⁴⁷⁴	
435	Original uploader was AverieZen ¹⁴⁷⁵ at en.wikibooks ¹⁴⁷⁶	
436	Original uploader was AverieZen ¹⁴⁷⁷ at en.wikibooks ¹⁴⁷⁸	
437	Original uploader was AverieZen ¹⁴⁷⁹ at en.wikibooks ¹⁴⁸⁰	
438	Original uploader was AverieZen ¹⁴⁸¹ at en.wikibooks ¹⁴⁸²	
439	Original uploader was AverieZen ¹⁴⁸³ at en.wikibooks ¹⁴⁸⁴	
440	DDD3x ¹⁴⁸⁵	PD
441	Telmer6 ¹⁴⁸⁶	cc-by-sa-3.0
442	Original uploader was Radialronnie ¹⁴⁸⁷	
443	Original uploader was Radialronnie ¹⁴⁸⁸	
444	pearts ¹⁴⁸⁹	
445	pearts ¹⁴⁹⁰	
446	Radialronnie ¹⁴⁹¹ . Original uploader was Radialronnie ¹⁴⁹² at en.wikibooks ¹⁴⁹³	
447	Original uploader was Radialronnie ¹⁴⁹⁴ at en.wikibooks ¹⁴⁹⁵	
448	pearts ¹⁴⁹⁶	
449	Original uploader was Radialronnie ¹⁴⁹⁷ at en.wikibooks ¹⁴⁹⁸	
450	Original uploader was Radialronnie ¹⁴⁹⁹ at en.wikibooks ¹⁵⁰⁰	
451	Original uploader was Radialronnie ¹⁵⁰¹ at en.wikibooks ¹⁵⁰²	

1471 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1472 <http://en.wikibooks.org>

1473 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1474 <http://en.wikibooks.org>

1475 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1476 <http://en.wikibooks.org>

1477 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1478 <http://en.wikibooks.org>

1479 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1480 <http://en.wikibooks.org>

1481 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1482 <http://en.wikibooks.org>

1483 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>

1484 <http://en.wikibooks.org>

1485 <http://en.wikibooks.org/wiki/User%3ADDD3x>

1486 <http://en.wikibooks.org/wiki/User%3ATelmer6>

1487 <http://en.wikibooks.org/wiki/User%3ARadialronnie>

1488 <http://en.wikibooks.org/wiki/User%3ARadialronnie>

1489 <http://en.wikibooks.org/wiki/User%3Apearts>

1490 <http://en.wikibooks.org/wiki/User%3Apearts>

1491 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

1492 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

1493 <http://en.wikibooks.org>

1494 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

1495 <http://en.wikibooks.org>

1496 <http://en.wikibooks.org/wiki/User%3Apearts>

1497 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

1498 <http://en.wikibooks.org>

1499 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

1500 <http://en.wikibooks.org>

1501 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

1502 <http://en.wikibooks.org>

452	Original uploader was Sumoncil ¹⁵⁰³ at en.wikibooks ¹⁵⁰⁴ . Later version(s) were uploaded by Adrignola ¹⁵⁰⁵ at en.wikibooks ¹⁵⁰⁶ .	
453	Original uploader was Sumoncil ¹⁵⁰⁷ at en.wikibooks ¹⁵⁰⁸ . Later version(s) were uploaded by Adrignola ¹⁵⁰⁹ at en.wikibooks ¹⁵¹⁰ .	
454	Original uploader was Sumoncil ¹⁵¹¹ at en.wikibooks ¹⁵¹² . Later version(s) were uploaded by Adrignola ¹⁵¹³ at en.wikibooks ¹⁵¹⁴ .	
455	Original uploader was Sumoncil ¹⁵¹⁵ at en.wikibooks ¹⁵¹⁶ . Later version(s) were uploaded by Adrignola ¹⁵¹⁷ at en.wikibooks ¹⁵¹⁸ .	
456	Original uploader was Sumoncil ¹⁵¹⁹ at en.wikibooks ¹⁵²⁰ . Later version(s) were uploaded by Adrignola ¹⁵²¹ at en.wikibooks ¹⁵²² .	
457	Original uploader was Sumoncil ¹⁵²³ at en.wikibooks ¹⁵²⁴ . Later version(s) were uploaded by Adrignola ¹⁵²⁵ at en.wikibooks ¹⁵²⁶ .	
458	Original uploader was Sumoncil ¹⁵²⁷ at en.wikibooks ¹⁵²⁸ . Later version(s) were uploaded by Adrignola ¹⁵²⁹ at en.wikibooks ¹⁵³⁰ .	
459	Original uploader was Sumoncil ¹⁵³¹ at en.wikibooks ¹⁵³² . Later version(s) were uploaded by Adrignola ¹⁵³³ at en.wikibooks ¹⁵³⁴ .	
460	Original uploader was Sumoncil ¹⁵³⁵ at en.wikibooks ¹⁵³⁶ . Later version(s) were uploaded by Adrignola ¹⁵³⁷ at en.wikibooks ¹⁵³⁸ .	

1503 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1504 <http://en.wikibooks.org>
1505 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1506 <http://en.wikibooks.org>
1507 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1508 <http://en.wikibooks.org>
1509 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1510 <http://en.wikibooks.org>
1511 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1512 <http://en.wikibooks.org>
1513 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1514 <http://en.wikibooks.org>
1515 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1516 <http://en.wikibooks.org>
1517 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1518 <http://en.wikibooks.org>
1519 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1520 <http://en.wikibooks.org>
1521 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1522 <http://en.wikibooks.org>
1523 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1524 <http://en.wikibooks.org>
1525 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1526 <http://en.wikibooks.org>
1527 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1528 <http://en.wikibooks.org>
1529 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1530 <http://en.wikibooks.org>
1531 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1532 <http://en.wikibooks.org>
1533 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1534 <http://en.wikibooks.org>
1535 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1536 <http://en.wikibooks.org>
1537 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1538 <http://en.wikibooks.org>

461	Original uploader was Sumoncil ¹⁵³⁹ at en.wikibooks ¹⁵⁴⁰ . Later version(s) were uploaded by Adrignola ¹⁵⁴¹ at en.wikibooks ¹⁵⁴² .	
462	Original uploader was Sumoncil ¹⁵⁴³ at en.wikibooks ¹⁵⁴⁴ . Later version(s) were uploaded by Adrignola ¹⁵⁴⁵ at en.wikibooks ¹⁵⁴⁶ .	
463	Original uploader was Sumoncil ¹⁵⁴⁷ at en.wikibooks ¹⁵⁴⁸ . Later version(s) were uploaded by Adrignola ¹⁵⁴⁹ at en.wikibooks ¹⁵⁵⁰ .	
464	Original uploader was Sumoncil ¹⁵⁵¹ at en.wikibooks ¹⁵⁵² . Later version(s) were uploaded by Adrignola ¹⁵⁵³ at en.wikibooks ¹⁵⁵⁴ .	
465	Original uploader was Sumoncil ¹⁵⁵⁵ at en.wikibooks ¹⁵⁵⁶ . Later version(s) were uploaded by Adrignola ¹⁵⁵⁷ at en.wikibooks ¹⁵⁵⁸ .	
466	Original uploader was Sumoncil ¹⁵⁵⁹ at en.wikibooks ¹⁵⁶⁰ . Later version(s) were uploaded by Adrignola ¹⁵⁶¹ at en.wikibooks ¹⁵⁶² .	
467	Original uploader was Sumoncil ¹⁵⁶³ at en.wikibooks ¹⁵⁶⁴ . Later version(s) were uploaded by Adrignola ¹⁵⁶⁵ at en.wikibooks ¹⁵⁶⁶ .	
468	Original uploader was Sumoncil ¹⁵⁶⁷ at en.wikibooks ¹⁵⁶⁸ . Later version(s) were uploaded by Adrignola ¹⁵⁶⁹ at en.wikibooks ¹⁵⁷⁰ .	
469	Original uploader was Sumoncil ¹⁵⁷¹ at en.wikibooks ¹⁵⁷² . Later version(s) were uploaded by Adrignola ¹⁵⁷³ at en.wikibooks ¹⁵⁷⁴ .	

1539 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1540 <http://en.wikibooks.org>
1541 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1542 <http://en.wikibooks.org>
1543 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1544 <http://en.wikibooks.org>
1545 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1546 <http://en.wikibooks.org>
1547 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1548 <http://en.wikibooks.org>
1549 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1550 <http://en.wikibooks.org>
1551 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1552 <http://en.wikibooks.org>
1553 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1554 <http://en.wikibooks.org>
1555 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1556 <http://en.wikibooks.org>
1557 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1558 <http://en.wikibooks.org>
1559 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1560 <http://en.wikibooks.org>
1561 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1562 <http://en.wikibooks.org>
1563 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1564 <http://en.wikibooks.org>
1565 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1566 <http://en.wikibooks.org>
1567 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1568 <http://en.wikibooks.org>
1569 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1570 <http://en.wikibooks.org>
1571 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
1572 <http://en.wikibooks.org>
1573 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
1574 <http://en.wikibooks.org>

470	Original uploader was Sumoncil ¹⁵⁷⁵ at en.wikibooks ¹⁵⁷⁶ . Later version(s) were uploaded by Adrignola ¹⁵⁷⁷ at en.wikibooks ¹⁵⁷⁸ .	
471	Original uploader was Sumoncil ¹⁵⁷⁹ at en.wikibooks ¹⁵⁸⁰ . Later version(s) were uploaded by Adrignola ¹⁵⁸¹ at en.wikibooks ¹⁵⁸² .	
472	Original uploader was Sumoncil ¹⁵⁸³ at en.wikibooks ¹⁵⁸⁴ . Later version(s) were uploaded by Adrignola ¹⁵⁸⁵ at en.wikibooks ¹⁵⁸⁶ .	
473	Original uploader was Sumoncil ¹⁵⁸⁷ at en.wikibooks ¹⁵⁸⁸ . Later version(s) were uploaded by Adrignola ¹⁵⁸⁹ at en.wikibooks ¹⁵⁹⁰ .	
474	Original uploader was Sumoncil ¹⁵⁹¹ at en.wikibooks ¹⁵⁹² . Later version(s) were uploaded by Adrignola ¹⁵⁹³ at en.wikibooks ¹⁵⁹⁴ .	
475	Original uploader was Sumoncil ¹⁵⁹⁵ at en.wikibooks ¹⁵⁹⁶ . Later version(s) were uploaded by Adrignola ¹⁵⁹⁷ at en.wikibooks ¹⁵⁹⁸ .	
476	Original uploader was Sumoncil ¹⁵⁹⁹ at en.wikibooks ¹⁶⁰⁰ . Later version(s) were uploaded by Adrignola ¹⁶⁰¹ at en.wikibooks ¹⁶⁰² .	
477	Original uploader was Sumoncil ¹⁶⁰³ at en.wikibooks ¹⁶⁰⁴ . Later version(s) were uploaded by Adrignola ¹⁶⁰⁵ at en.wikibooks ¹⁶⁰⁶ .	
478	Original uploader was Sumoncil ¹⁶⁰⁷ at en.wikibooks ¹⁶⁰⁸	

[1575 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1576 http://en.wikibooks.org](http://en.wikibooks.org)
[1577 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1578 http://en.wikibooks.org](http://en.wikibooks.org)
[1579 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1580 http://en.wikibooks.org](http://en.wikibooks.org)
[1581 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1582 http://en.wikibooks.org](http://en.wikibooks.org)
[1583 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1584 http://en.wikibooks.org](http://en.wikibooks.org)
[1585 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1586 http://en.wikibooks.org](http://en.wikibooks.org)
[1587 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1588 http://en.wikibooks.org](http://en.wikibooks.org)
[1589 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1590 http://en.wikibooks.org](http://en.wikibooks.org)
[1591 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1592 http://en.wikibooks.org](http://en.wikibooks.org)
[1593 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1594 http://en.wikibooks.org](http://en.wikibooks.org)
[1595 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1596 http://en.wikibooks.org](http://en.wikibooks.org)
[1597 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1598 http://en.wikibooks.org](http://en.wikibooks.org)
[1599 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1600 http://en.wikibooks.org](http://en.wikibooks.org)
[1601 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1602 http://en.wikibooks.org](http://en.wikibooks.org)
[1603 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1604 http://en.wikibooks.org](http://en.wikibooks.org)
[1605 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1606 http://en.wikibooks.org](http://en.wikibooks.org)
[1607 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1608 http://en.wikibooks.org](http://en.wikibooks.org)

479	Original uploader was Sumoncil ¹⁶⁰⁹ at en.wikibooks ¹⁶¹⁰ . Later version(s) were uploaded by Adrignola ¹⁶¹¹ at en.wikibooks ¹⁶¹² .	
480	Original uploader was Sumoncil ¹⁶¹³ at en.wikibooks ¹⁶¹⁴ . Later version(s) were uploaded by Adrignola ¹⁶¹⁵ at en.wikibooks ¹⁶¹⁶ .	
481	Original uploader was Sumoncil ¹⁶¹⁷ at en.wikibooks ¹⁶¹⁸ . Later version(s) were uploaded by Adrignola ¹⁶¹⁹ at en.wikibooks ¹⁶²⁰ .	
482	Original uploader was Sumoncil ¹⁶²¹ at en.wikibooks ¹⁶²² . Later version(s) were uploaded by Adrignola ¹⁶²³ at en.wikibooks ¹⁶²⁴ .	
483	Original uploader was Sumoncil ¹⁶²⁵ at en.wikibooks ¹⁶²⁶ . Later version(s) were uploaded by Adrignola ¹⁶²⁷ at en.wikibooks ¹⁶²⁸ .	
484	Original uploader was Sumoncil ¹⁶²⁹ at en.wikibooks ¹⁶³⁰ . Later version(s) were uploaded by Adrignola ¹⁶³¹ at en.wikibooks ¹⁶³² .	
485	Original uploader was Sumoncil ¹⁶³³ at en.wikibooks ¹⁶³⁴ . Later version(s) were uploaded by Adrignola ¹⁶³⁵ at en.wikibooks ¹⁶³⁶ .	
486	Original uploader was Sumoncil ¹⁶³⁷ at en.wikibooks ¹⁶³⁸ . Later version(s) were uploaded by Adrignola ¹⁶³⁹ at en.wikibooks ¹⁶⁴⁰ .	
487	Original uploader was Wynnmc ¹⁶⁴¹ at en.wikibooks ¹⁶⁴² .	
488	Original uploader was Wynnmc ¹⁶⁴³ at en.wikibooks ¹⁶⁴⁴ .	

[1609 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1610 http://en.wikibooks.org](http://en.wikibooks.org)
[1611 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1612 http://en.wikibooks.org](http://en.wikibooks.org)
[1613 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1614 http://en.wikibooks.org](http://en.wikibooks.org)
[1615 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1616 http://en.wikibooks.org](http://en.wikibooks.org)
[1617 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1618 http://en.wikibooks.org](http://en.wikibooks.org)
[1619 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1620 http://en.wikibooks.org](http://en.wikibooks.org)
[1621 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1622 http://en.wikibooks.org](http://en.wikibooks.org)
[1623 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1624 http://en.wikibooks.org](http://en.wikibooks.org)
[1625 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1626 http://en.wikibooks.org](http://en.wikibooks.org)
[1627 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1628 http://en.wikibooks.org](http://en.wikibooks.org)
[1629 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1630 http://en.wikibooks.org](http://en.wikibooks.org)
[1631 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1632 http://en.wikibooks.org](http://en.wikibooks.org)
[1633 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1634 http://en.wikibooks.org](http://en.wikibooks.org)
[1635 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1636 http://en.wikibooks.org](http://en.wikibooks.org)
[1637 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[1638 http://en.wikibooks.org](http://en.wikibooks.org)
[1639 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[1640 http://en.wikibooks.org](http://en.wikibooks.org)
[1641 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc)
[1642 http://en.wikibooks.org](http://en.wikibooks.org)
[1643 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc)
[1644 http://en.wikibooks.org](http://en.wikibooks.org)

489	pearts ¹⁶⁴⁵	
490	Original uploader was Wynnmc ¹⁶⁴⁶ at en.wikibooks ¹⁶⁴⁷	
491	Original uploader was Wynnmc ¹⁶⁴⁸ at en.wikibooks ¹⁶⁴⁹	
492	Original uploader was Wynnmc ¹⁶⁵⁰ at en.wikibooks ¹⁶⁵¹	
493	Original uploader was Wynnmc ¹⁶⁵² at en.wikibooks ¹⁶⁵³	
494	Original uploader was Wynnmc ¹⁶⁵⁴ at en.wikibooks ¹⁶⁵⁵	
495	Original uploader was Wynnmc ¹⁶⁵⁶ at en.wikibooks ¹⁶⁵⁷	
496	Original uploader was Wynnmc ¹⁶⁵⁸ at en.wikibooks ¹⁶⁵⁹	
497	Original uploader was Wynnmc ¹⁶⁶⁰ at en.wikibooks ¹⁶⁶¹	
498	Original uploader was Wynnmc ¹⁶⁶² at en.wikibooks ¹⁶⁶³	
499	pearts ¹⁶⁶⁴	
500	Original uploader was Wynnmc ¹⁶⁶⁵ at en.wikibooks ¹⁶⁶⁶	
501	Original uploader was Wynnmc ¹⁶⁶⁷ at en.wikibooks ¹⁶⁶⁸	
0	Loremus Ipsemus	None
503	pearts ¹⁶⁶⁹	
504	pearts ¹⁶⁷⁰	
505	pearts ¹⁶⁷¹	
506	pearts ¹⁶⁷²	
507	Original uploader was Wynnmc ¹⁶⁷³ at en.wikibooks ¹⁶⁷⁴	
508	pearts ¹⁶⁷⁵	
509	pearts ¹⁶⁷⁶	

-
- 1645 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1646 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1647 <http://en.wikibooks.org>
 - 1648 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1649 <http://en.wikibooks.org>
 - 1650 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1651 <http://en.wikibooks.org>
 - 1652 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1653 <http://en.wikibooks.org>
 - 1654 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1655 <http://en.wikibooks.org>
 - 1656 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1657 <http://en.wikibooks.org>
 - 1658 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1659 <http://en.wikibooks.org>
 - 1660 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1661 <http://en.wikibooks.org>
 - 1662 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1663 <http://en.wikibooks.org>
 - 1664 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1665 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1666 <http://en.wikibooks.org>
 - 1667 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1668 <http://en.wikibooks.org>
 - 1669 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1670 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1671 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1672 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1673 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
 - 1674 <http://en.wikibooks.org>
 - 1675 <http://en.wikibooks.org/wiki/User%3Apearts>
 - 1676 <http://en.wikibooks.org/wiki/User%3Apearts>

510	pearts ¹⁶⁷⁷	
511	pearts ¹⁶⁷⁸	
512	pearts ¹⁶⁷⁹	
513	pearts ¹⁶⁸⁰	
514	pearts ¹⁶⁸¹	
515	Original uploader was Wynnmc ¹⁶⁸² at en.wikibooks ¹⁶⁸³	
0	Loremus Ipsems	None
517	Original uploader was Wynnmc ¹⁶⁸⁴ at en.wikibooks ¹⁶⁸⁵	
518	Original uploader was Wynnmc ¹⁶⁸⁶ at en.wikibooks ¹⁶⁸⁷	
519	pearts ¹⁶⁸⁸	
520	SoylentGreen ¹⁶⁸⁹	GFDL
521	SoylentGreen ¹⁶⁹⁰	GFDL
522	SoylentGreen ¹⁶⁹¹	GFDL
523	SoylentGreen ¹⁶⁹²	GFDL
524	SoylentGreen ¹⁶⁹³	GFDL
525		GFDL
526	Lawrence D'Oliveiro	GPL
527	pearts ¹⁶⁹⁴	
528		PD
529	Original uploader was Lirtferk ¹⁶⁹⁵ at en.wikibooks ¹⁶⁹⁶	PD
530		cc-by-sa-2.5
531	Original uploader was Badalia ¹⁶⁹⁷ at en.wikibooks ¹⁶⁹⁸	cc-by-sa-2.5
532	Original uploader was Badalia ¹⁶⁹⁹ at en.wikibooks ¹⁷⁰⁰	cc-by-sa-2.5
533	Original uploader was Badalia ¹⁷⁰¹ at en.wikibooks ¹⁷⁰²	cc-by-sa-2.5
534		GFDL
535		cc-by-sa-2.5

1677 <http://en.wikibooks.org/wiki/User%3Apearts>
1678 <http://en.wikibooks.org/wiki/User%3Apearts>
1679 <http://en.wikibooks.org/wiki/User%3Apearts>
1680 <http://en.wikibooks.org/wiki/User%3Apearts>
1681 <http://en.wikibooks.org/wiki/User%3Apearts>
1682 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1683 <http://en.wikibooks.org>
1684 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1685 <http://en.wikibooks.org>
1686 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWynnmc>
1687 <http://en.wikibooks.org>
1688 <http://en.wikibooks.org/wiki/User%3Apearts>
1689 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
1690 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
1691 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
1692 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
1693 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
1694 <http://en.wikibooks.org/wiki/User%3Apearts>
1695 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALirtferk>
1696 <http://en.wikibooks.org>
1697 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABadalia>
1698 <http://en.wikibooks.org>
1699 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABadalia>
1700 <http://en.wikibooks.org>
1701 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABadalia>
1702 <http://en.wikibooks.org>

536		cc-by-sa-2.5
537		cc-by-sa-2.5
538		cc-by-sa-2.5
539		cc-by-sa-2.5
540		cc-by-sa-2.5
541		cc-by-sa-2.5
542	Original uploader was AverieZen ¹⁷⁰³ at en.wikibooks ¹⁷⁰⁴	
543	Original uploader was AverieZen ¹⁷⁰⁵ at en.wikibooks ¹⁷⁰⁶	
544	Original uploader was AverieZen ¹⁷⁰⁷ at en.wikibooks ¹⁷⁰⁸	
545	Original uploader was AverieZen ¹⁷⁰⁹ at en.wikibooks ¹⁷¹⁰	
546	Original uploader was AverieZen ¹⁷¹¹ at en.wikibooks ¹⁷¹²	
547	Original uploader was AverieZen ¹⁷¹³ at en.wikibooks ¹⁷¹⁴	
548	Original uploader was AverieZen ¹⁷¹⁵ at en.wikibooks ¹⁷¹⁶	
549	Original uploader was AverieZen ¹⁷¹⁷ at en.wikibooks ¹⁷¹⁸	
550		GFDL
551		GFDL
552		GFDL
553		GFDL
554	Original uploader was Mrfelis ¹⁷¹⁹ at en.wikibooks ¹⁷²⁰	
555	Original uploader was Yoyotweak ¹⁷²¹ at en.wikibooks ¹⁷²²	
556	Original uploader was Yoyotweak ¹⁷²³ at en.wikibooks ¹⁷²⁴	
557	Original uploader was Yoyotweak ¹⁷²⁵ at en.wikibooks ¹⁷²⁶	
558	Original uploader was Yoyotweak ¹⁷²⁷ at en.wikibooks ¹⁷²⁸	
559	Original uploader was Yoyotweak ¹⁷²⁹ at en.wikibooks ¹⁷³⁰	

1703 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1704 <http://en.wikibooks.org>
1705 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1706 <http://en.wikibooks.org>
1707 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1708 <http://en.wikibooks.org>
1709 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1710 <http://en.wikibooks.org>
1711 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1712 <http://en.wikibooks.org>
1713 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1714 <http://en.wikibooks.org>
1715 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1716 <http://en.wikibooks.org>
1717 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen>
1718 <http://en.wikibooks.org>
1719 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>
1720 <http://en.wikibooks.org>
1721 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>
1722 <http://en.wikibooks.org>
1723 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>
1724 <http://en.wikibooks.org>
1725 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>
1726 <http://en.wikibooks.org>
1727 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>
1728 <http://en.wikibooks.org>
1729 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>
1730 <http://en.wikibooks.org>

560	Original uploader was Yoyotweak ¹⁷³¹ at en.wikibooks ¹⁷³²	
561	Original uploader was Yoyotweak ¹⁷³³ at en.wikibooks ¹⁷³⁴	
562	Original uploader was Yoyotweak ¹⁷³⁵ at en.wikibooks ¹⁷³⁶	
563	Original uploader was Yoyotweak ¹⁷³⁷ at en.wikibooks ¹⁷³⁸	
564	Original uploader was Yoyotweak ¹⁷³⁹ at en.wikibooks ¹⁷⁴⁰	
565	Original uploader was Mrfelis ¹⁷⁴¹ at en.wikibooks ¹⁷⁴²	
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
569	Original uploader was Mrfelis ¹⁷⁴³ at en.wikibooks ¹⁷⁴⁴	
570	Original uploader was Mrfelis ¹⁷⁴⁵ at en.wikibooks ¹⁷⁴⁶	
571	Original uploader was Mrfelis ¹⁷⁴⁷ at en.wikibooks ¹⁷⁴⁸	
572	Original uploader was Mrfelis ¹⁷⁴⁹ at en.wikibooks ¹⁷⁵⁰	
573	Mrfelis ¹⁷⁵¹ . Original uploader was Mrfelis ¹⁷⁵² at en.wikibooks ¹⁷⁵³	
574	Original uploader was Mrfelis ¹⁷⁵⁴ at en.wikibooks ¹⁷⁵⁵	
575	Mrfelis ¹⁷⁵⁶ . Original uploader was Mrfelis ¹⁷⁵⁷ at en.wikibooks ¹⁷⁵⁸	
576	Mrfelis ¹⁷⁵⁹ . Original uploader was Mrfelis ¹⁷⁶⁰ at en.wikibooks ¹⁷⁶¹	

1731 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>

1732 <http://en.wikibooks.org>

1733 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>

1734 <http://en.wikibooks.org>

1735 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>

1736 <http://en.wikibooks.org>

1737 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>

1738 <http://en.wikibooks.org>

1739 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AYoyotweak>

1740 <http://en.wikibooks.org>

1741 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1742 <http://en.wikibooks.org>

1743 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1744 <http://en.wikibooks.org>

1745 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1746 <http://en.wikibooks.org>

1747 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1748 <http://en.wikibooks.org>

1749 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1750 <http://en.wikibooks.org>

1751 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1752 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1753 <http://en.wikibooks.org>

1754 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1755 <http://en.wikibooks.org>

1756 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1757 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1758 <http://en.wikibooks.org>

1759 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1760 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1761 <http://en.wikibooks.org>

577	Mrfelis ¹⁷⁶² . Original uploader was Mrfelis ¹⁷⁶³ at en.wikibooks ¹⁷⁶⁴	
578	pearts ¹⁷⁶⁵	
579	pearts ¹⁷⁶⁶	
580	Original uploader was Kieran ¹⁷⁶⁷ at en.wikibooks ¹⁷⁶⁸	GFDL
581	Original uploader was Kieran ¹⁷⁶⁹ at en.wikibooks ¹⁷⁷⁰	GFDL
582	Original uploader was Kieran ¹⁷⁷¹ at en.wikibooks ¹⁷⁷²	GFDL
583	Original uploader was Kieran ¹⁷⁷³ at en.wikibooks ¹⁷⁷⁴	GFDL
584	Original uploader was Kieran ¹⁷⁷⁵ at en.wikibooks ¹⁷⁷⁶	GFDL
585	Original uploader was Kieran ¹⁷⁷⁷ at en.wikibooks ¹⁷⁷⁸	GFDL
586	Original uploader was Kieran ¹⁷⁷⁹ at en.wikibooks ¹⁷⁸⁰	GFDL
587	pearts ¹⁷⁸¹	
588	Original uploader was Kieran ¹⁷⁸² at en.wikibooks ¹⁷⁸³	GFDL
589	Original uploader was Kieran ¹⁷⁸⁴ at en.wikibooks ¹⁷⁸⁵	GFDL
590	pearts ¹⁷⁸⁶	
591	Original uploader was Kieran ¹⁷⁸⁷ at en.wikibooks ¹⁷⁸⁸	GFDL
592	Original uploader was Johno ¹⁷⁸⁹ at en.wikibooks ¹⁷⁹⁰	
593	Original uploader was Johno ¹⁷⁹¹ at en.wikibooks ¹⁷⁹²	
594	Original uploader was Johno ¹⁷⁹³ at en.wikibooks ¹⁷⁹⁴	

1762 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1763 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

1764 <http://en.wikibooks.org>

1765 <http://en.wikibooks.org/wiki/User%3Apearts>

1766 <http://en.wikibooks.org/wiki/User%3Apearts>

1767 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1768 <http://en.wikibooks.org>

1769 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1770 <http://en.wikibooks.org>

1771 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1772 <http://en.wikibooks.org>

1773 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1774 <http://en.wikibooks.org>

1775 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1776 <http://en.wikibooks.org>

1777 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1778 <http://en.wikibooks.org>

1779 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1780 <http://en.wikibooks.org>

1781 <http://en.wikibooks.org/wiki/User%3Apearts>

1782 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1783 <http://en.wikibooks.org>

1784 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1785 <http://en.wikibooks.org>

1786 <http://en.wikibooks.org/wiki/User%3Apearts>

1787 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AKieran>

1788 <http://en.wikibooks.org>

1789 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohno>

1790 <http://en.wikibooks.org>

1791 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohno>

1792 <http://en.wikibooks.org>

1793 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohno>

1794 <http://en.wikibooks.org>

595	Original uploader was Johno ¹⁷⁹⁵ at en.wikibooks ¹⁷⁹⁶	
596		PD
597	Original uploader was Johno ¹⁷⁹⁷ at en.wikibooks ¹⁷⁹⁸	
598	Original uploader was Darkonc ¹⁷⁹⁹ at en.wikibooks ¹⁸⁰⁰	GFDL
599		GFDL
600	Original uploader was Darkonc ¹⁸⁰¹ at en.wikibooks ¹⁸⁰²	GFDL
601		GFDL
602		GFDL
603		GFDL
604		GFDL
605	Original uploader was Lazy-lump ¹⁸⁰³ at en.wikibooks ¹⁸⁰⁴	PD
606	Original uploader was Lazy-lump ¹⁸⁰⁵ at en.wikibooks ¹⁸⁰⁶	PD
607	Original uploader was Lazy-lump ¹⁸⁰⁷ at en.wikibooks ¹⁸⁰⁸	
608	Original uploader was Lazy-lump ¹⁸⁰⁹ at en.wikibooks ¹⁸¹⁰	PD
609	Original uploader was Lazy-lump ¹⁸¹¹ at en.wikibooks ¹⁸¹²	PD
610	Original uploader was Lazy-lump ¹⁸¹³ at en.wikibooks ¹⁸¹⁴	PD
611	Original uploader was Lazy-lump ¹⁸¹⁵ at en.wikibooks ¹⁸¹⁶	
612		PD
613	Original uploader was Embri ¹⁸¹⁷ at en.wikibooks ¹⁸¹⁸	
614	Original uploader was Embri ¹⁸¹⁹ at en.wikibooks ¹⁸²⁰	
615	Original uploader was Embri ¹⁸²¹ at en.wikibooks ¹⁸²²	
616		
617	Embri. Original uploader was Embri ¹⁸²³ at en.wikibooks ¹⁸²⁴	

¹⁷⁹⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohno>

¹⁷⁹⁶ <http://en.wikibooks.org>

¹⁷⁹⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohno>

¹⁷⁹⁸ <http://en.wikibooks.org>

¹⁷⁹⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

¹⁸⁰⁰ <http://en.wikibooks.org>

¹⁸⁰¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

¹⁸⁰² <http://en.wikibooks.org>

¹⁸⁰³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸⁰⁴ <http://en.wikibooks.org>

¹⁸⁰⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸⁰⁶ <http://en.wikibooks.org>

¹⁸⁰⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸⁰⁸ <http://en.wikibooks.org>

¹⁸⁰⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸¹⁰ <http://en.wikibooks.org>

¹⁸¹¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸¹² <http://en.wikibooks.org>

¹⁸¹³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸¹⁴ <http://en.wikibooks.org>

¹⁸¹⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALazy-lump>

¹⁸¹⁶ <http://en.wikibooks.org>

¹⁸¹⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AEmbri>

¹⁸¹⁸ <http://en.wikibooks.org>

¹⁸¹⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AEmbri>

¹⁸²⁰ <http://en.wikibooks.org>

¹⁸²¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AEmbri>

¹⁸²² <http://en.wikibooks.org>

¹⁸²³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AEmbri>

¹⁸²⁴ <http://en.wikibooks.org>

618	pearts ¹⁸²⁵	
619	pearts ¹⁸²⁶	
620	pearts ¹⁸²⁷	
621	SoylentGreen ¹⁸²⁸	GFDL
622	SoylentGreen ¹⁸²⁹	
623	SoylentGreen ¹⁸³⁰	GPL
624	SoylentGreen ¹⁸³¹	GFDL
625	SoylentGreen ¹⁸³²	GFDL
626	SoylentGreen ¹⁸³³ using a file from the releasenotes of Blender version 2.32	GFDL
627	SoylentGreen ¹⁸³⁴ using a file from the releasenotes of Blender version 2.32	GFDL
628	SoylentGreen ¹⁸³⁵	GPL
629	SoylentGreen ¹⁸³⁶	GPL
630	SoylentGreen ¹⁸³⁷	GPL
631	SoylentGreen ¹⁸³⁸	GFDL
632	SoylentGreen ¹⁸³⁹	GFDL
633	SoylentGreen ¹⁸⁴⁰	GFDL
634	SoylentGreen ¹⁸⁴¹	GFDL
635	SoylentGreen ¹⁸⁴²	PD
636	SoylentGreen ¹⁸⁴³	GPL
637	SoylentGreen ¹⁸⁴⁴	
638	SoylentGreen ¹⁸⁴⁵	GPL
639	SoylentGreen ¹⁸⁴⁶	PD
640	SoylentGreen ¹⁸⁴⁷ , Krzysztof P. Jasiutowicz	GFDL
641	SoylentGreen ¹⁸⁴⁸ (Edited by: Weeix ¹⁸⁴⁹)	GPL

1825 <http://en.wikibooks.org/wiki/User%3Apearts>
 1826 <http://en.wikibooks.org/wiki/User%3Apearts>
 1827 <http://en.wikibooks.org/wiki/User%3Apearts>
 1828 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1829 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1830 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1831 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1832 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1833 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1834 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1835 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1836 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1837 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1838 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1839 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1840 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1841 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1842 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1843 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1844 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1845 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1846 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1847 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1848 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 1849 <http://en.wikibooks.org/wiki/User%3AWeeix>

642	SoylentGreen ¹⁸⁵⁰	GFDL
643	Weeix ¹⁸⁵¹ (talk ¹⁸⁵²) 14:15, 2 November 2010 (UTC)	GPL
644	Dobz116 ¹⁸⁵³	GFDL
645		GFDL
646		PD
647	Original uploader was Dan13 ¹⁸⁵⁴ at en.wikibooks ¹⁸⁵⁵	GFDL
648		cc-by-sa-2.5
649	en:Dan13 ¹⁸⁵⁶	
650		cc-by-sa-2.5
651		cc-by-sa-2.5
652		cc-by-sa-2.5
653		cc-by-sa-2.5
654		cc-by-sa-2.5
655		cc-by-sa-2.5
656		cc-by-sa-2.5
657		cc-by-sa-2.5
658		cc-by-sa-2.5
659		cc-by-sa-2.5
660		cc-by-sa-2.5
661	Weeix ¹⁸⁵⁷	
662	Original uploader was Hrakaroo ¹⁸⁵⁸ at en.wikibooks ¹⁸⁵⁹	PD
663	Original uploader was WeirdHat ¹⁸⁶⁰ at en.wikibooks ¹⁸⁶¹ . Later version(s) were uploaded by Rdp ¹⁸⁶² at en.wikibooks ¹⁸⁶³ .	PD
664	Original uploader was Icky-wiki ¹⁸⁶⁴ at en.wikibooks ¹⁸⁶⁵	GPL
665		
666	Original uploader was Hrakaroo ¹⁸⁶⁶ at en.wikibooks ¹⁸⁶⁷	GPL
667	Original uploader was ThomasB ¹⁸⁶⁸ at en.wikibooks ¹⁸⁶⁹	GPL
668	Original uploader was Chinmay gautam ¹⁸⁷⁰ at en.wikibooks ¹⁸⁷¹	cc-by-sa-2.5

1850 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
1851 <http://en.wikibooks.org/wiki/User%3AWeeix>
1852 <http://en.wikibooks.org/wiki/User%20talk%3AWeeix>
1853 <http://en.wikibooks.org/wiki/User%3ADobz116>
1854 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADan13>
1855 <http://en.wikibooks.org>
1856 <http://en.wikibooks.org/wiki/Dan13>
1857 <http://en.wikibooks.org/wiki/User%3AWeeix>
1858 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo>
1859 <http://en.wikibooks.org>
1860 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AWeirdHat>
1861 <http://en.wikibooks.org>
1862 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARdp>
1863 <http://en.wikibooks.org>
1864 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AIcky-wiki>
1865 <http://en.wikibooks.org>
1866 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo>
1867 <http://en.wikibooks.org>
1868 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AThomasB>
1869 <http://en.wikibooks.org>
1870 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam>
1871 <http://en.wikibooks.org>

669	Original uploader was Chinmay gautam ¹⁸⁷² at en.wikibooks ¹⁸⁷³	cc-by-sa-2.5
670	Original uploader was Chinmay gautam ¹⁸⁷⁴ at en.wikibooks ¹⁸⁷⁵	cc-by-sa-2.5
671	Original uploader was Chinmay gautam ¹⁸⁷⁶ at en.wikibooks ¹⁸⁷⁷	cc-by-sa-2.5
672	Original uploader was Hrakaroo ¹⁸⁷⁸ at en.wikibooks ¹⁸⁷⁹	GPL
673	Original uploader was Hrakaroo ¹⁸⁸⁰ at en.wikibooks ¹⁸⁸¹	PD
674	Original uploader was Hrakaroo ¹⁸⁸² at en.wikibooks ¹⁸⁸³	PD
675	Original uploader was Hrakaroo ¹⁸⁸⁴ at en.wikibooks ¹⁸⁸⁵	PD
676	Original uploader was ZeroOne ¹⁸⁸⁶ at en.wikibooks ¹⁸⁸⁷	
677	Original uploader was ZeroOne ¹⁸⁸⁸ at en.wikibooks ¹⁸⁸⁹	
678	Original uploader was ZeroOne ¹⁸⁹⁰ at en.wikibooks ¹⁸⁹¹	
679	Original uploader was ZeroOne ¹⁸⁹² at en.wikibooks ¹⁸⁹³	
680		PD
681	Original uploader was ZeroOne ¹⁸⁹⁴ at en.wikibooks ¹⁸⁹⁵	
682	Original uploader was ZeroOne ¹⁸⁹⁶ at en.wikibooks ¹⁸⁹⁷	
683	Original uploader was ZeroOne ¹⁸⁹⁸ at en.wikibooks ¹⁸⁹⁹	
684	Original uploader was Argento ¹⁹⁰⁰ at en.wikibooks ¹⁹⁰¹	
685	Othmanskn2 ¹⁹⁰²	cc-by-sa-3.0
686	Othmanskn2 ¹⁹⁰³	cc-by-sa-3.0
687	Othmanskn2 ¹⁹⁰⁴	cc-by-sa-3.0

[1872 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam)
[1873 http://en.wikibooks.org](http://en.wikibooks.org)
[1874 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam)
[1875 http://en.wikibooks.org](http://en.wikibooks.org)
[1876 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AChinmay%20gautam)
[1877 http://en.wikibooks.org](http://en.wikibooks.org)
[1878 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo)
[1879 http://en.wikibooks.org](http://en.wikibooks.org)
[1880 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo)
[1881 http://en.wikibooks.org](http://en.wikibooks.org)
[1882 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo)
[1883 http://en.wikibooks.org](http://en.wikibooks.org)
[1884 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHrakaroo)
[1885 http://en.wikibooks.org](http://en.wikibooks.org)
[1886 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1887 http://en.wikibooks.org](http://en.wikibooks.org)
[1888 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1889 http://en.wikibooks.org](http://en.wikibooks.org)
[1890 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1891 http://en.wikibooks.org](http://en.wikibooks.org)
[1892 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1893 http://en.wikibooks.org](http://en.wikibooks.org)
[1894 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1895 http://en.wikibooks.org](http://en.wikibooks.org)
[1896 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1897 http://en.wikibooks.org](http://en.wikibooks.org)
[1898 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[1899 http://en.wikibooks.org](http://en.wikibooks.org)
[1900 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento)
[1901 http://en.wikibooks.org](http://en.wikibooks.org)
[1902 http://en.wikibooks.org/wiki/User%3AOthmanskn2](http://en.wikibooks.org/wiki/User%3AOthmanskn2)
[1903 http://en.wikibooks.org/wiki/User%3AOthmanskn2](http://en.wikibooks.org/wiki/User%3AOthmanskn2)
[1904 http://en.wikibooks.org/wiki/User%3AOthmanskn2](http://en.wikibooks.org/wiki/User%3AOthmanskn2)

688	Othmanskn2 ¹⁹⁰⁵	cc-by-sa-3.0
0	Loremus Ipsemus	None
690		PD
691		PD
692		PD
693	Original uploader was Quantum Anomaly ¹⁹⁰⁶ at en.wikibooks ¹⁹⁰⁷	
694		PD
695		PD
696		GFDL
697		PD
698		PD
699		PD
700		PD
701	Original uploader was Quantum Anomaly ¹⁹⁰⁸ at en.wikibooks ¹⁹⁰⁹	PD
0	Loremus Ipsemus	None
703		PD
704	Original uploader was Quantum Anomaly ¹⁹¹⁰ at en.wikibooks ¹⁹¹¹	PD
705	Original uploader was Quantum Anomaly ¹⁹¹² at en.wikibooks ¹⁹¹³	PD
706	Original uploader was Quantum Anomaly ¹⁹¹⁴ at en.wikibooks ¹⁹¹⁵	PD
707	Original uploader was Quantum Anomaly ¹⁹¹⁶ at en.wikibooks ¹⁹¹⁷	PD
708	Original uploader was Quantum Anomaly ¹⁹¹⁸ at en.wikibooks ¹⁹¹⁹	
709	Weeix ¹⁹²⁰	
710	Weeix ¹⁹²¹	
711	Weeix ¹⁹²²	
712	Weeix ¹⁹²³	
713	Weeix ¹⁹²⁴	
714		PD
715	Original uploader was Totophe64 ¹⁹²⁵ at en.wikibooks ¹⁹²⁶	

1905 <http://en.wikibooks.org/wiki/User%3AOthmanskn2>

1906 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1907 <http://en.wikibooks.org>

1908 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1909 <http://en.wikibooks.org>

1910 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1911 <http://en.wikibooks.org>

1912 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1913 <http://en.wikibooks.org>

1914 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1915 <http://en.wikibooks.org>

1916 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1917 <http://en.wikibooks.org>

1918 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

1919 <http://en.wikibooks.org>

1920 <http://en.wikibooks.org/wiki/User%3AWeeix>

1921 <http://en.wikibooks.org/wiki/User%3AWeeix>

1922 <http://en.wikibooks.org/wiki/User%3AWeeix>

1923 <http://en.wikibooks.org/wiki/User%3AWeeix>

1924 <http://en.wikibooks.org/wiki/User%3AWeeix>

1925 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1926 <http://en.wikibooks.org>

716	Original uploader was Totophe64 ¹⁹²⁷ at en.wikibooks ¹⁹²⁸	
717	Original uploader was Totophe64 ¹⁹²⁹ at en.wikibooks ¹⁹³⁰	
718	Original uploader was Totophe64 ¹⁹³¹ at en.wikibooks ¹⁹³²	
719	Original uploader was Totophe64 ¹⁹³³ at en.wikibooks ¹⁹³⁴	
720	Original uploader was Totophe64 ¹⁹³⁵ at en.wikibooks ¹⁹³⁶	
721	Original uploader was Totophe64 ¹⁹³⁷ at en.wikibooks ¹⁹³⁸	
722	Original uploader was Totophe64 ¹⁹³⁹ at en.wikibooks ¹⁹⁴⁰	
723	Original uploader was Totophe64 ¹⁹⁴¹ at en.wikibooks ¹⁹⁴²	
0	Loremus Ipsemus	None
725	Original uploader was Totophe64 ¹⁹⁴³ at en.wikibooks ¹⁹⁴⁴	GFDL
726	Original uploader was Totophe64 ¹⁹⁴⁵ at en.wikibooks ¹⁹⁴⁶	GFDL
727	Original uploader was Totophe64 ¹⁹⁴⁷ at en.wikibooks ¹⁹⁴⁸	
728	Original uploader was Totophe64 ¹⁹⁴⁹ at en.wikibooks ¹⁹⁵⁰	GFDL
729		GFDL
730		GFDL
731		GFDL
732		GFDL
733		GFDL
734		GFDL
735	Original uploader was Soeb ¹⁹⁵¹ at en.wikibooks ¹⁹⁵²	
736	Original uploader was Soeb ¹⁹⁵³ at en.wikibooks ¹⁹⁵⁴	
737	Original uploader was Soeb ¹⁹⁵⁵ at en.wikibooks ¹⁹⁵⁶	

1927 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1928 <http://en.wikibooks.org>

1929 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1930 <http://en.wikibooks.org>

1931 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1932 <http://en.wikibooks.org>

1933 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1934 <http://en.wikibooks.org>

1935 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1936 <http://en.wikibooks.org>

1937 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1938 <http://en.wikibooks.org>

1939 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1940 <http://en.wikibooks.org>

1941 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1942 <http://en.wikibooks.org>

1943 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1944 <http://en.wikibooks.org>

1945 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1946 <http://en.wikibooks.org>

1947 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1948 <http://en.wikibooks.org>

1949 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

1950 <http://en.wikibooks.org>

1951 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1952 <http://en.wikibooks.org>

1953 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1954 <http://en.wikibooks.org>

1955 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1956 <http://en.wikibooks.org>

738	Original uploader was Soeb ¹⁹⁵⁷ at en.wikibooks ¹⁹⁵⁸	
739	Original uploader was Soeb ¹⁹⁵⁹ at en.wikibooks ¹⁹⁶⁰	
740	Original uploader was Soeb ¹⁹⁶¹ at en.wikibooks ¹⁹⁶²	
741	Original uploader was Soeb ¹⁹⁶³ at en.wikibooks ¹⁹⁶⁴	
742	Original uploader was Soeb ¹⁹⁶⁵ at en.wikibooks ¹⁹⁶⁶	
743	Original uploader was Soeb ¹⁹⁶⁷ at en.wikibooks ¹⁹⁶⁸	
744	Original uploader was Soeb ¹⁹⁶⁹ at en.wikibooks ¹⁹⁷⁰	
745	Original uploader was Soeb ¹⁹⁷¹ at en.wikibooks ¹⁹⁷²	
746	Original uploader was Soeb ¹⁹⁷³ at en.wikibooks ¹⁹⁷⁴	
747	Original uploader was Soeb ¹⁹⁷⁵ at en.wikibooks ¹⁹⁷⁶	
748	Original uploader was Soeb ¹⁹⁷⁷ at en.wikibooks ¹⁹⁷⁸	
749	Original uploader was Soeb ¹⁹⁷⁹ at en.wikibooks ¹⁹⁸⁰	
750	Original uploader was Soeb ¹⁹⁸¹ at en.wikibooks ¹⁹⁸²	
751	Original uploader was Soeb ¹⁹⁸³ at en.wikibooks ¹⁹⁸⁴	
752	Original uploader was Soeb ¹⁹⁸⁵ at en.wikibooks ¹⁹⁸⁶	
753		GFDL
754	Original uploader was Allefant ¹⁹⁸⁷ at en.wikibooks ¹⁹⁸⁸	
755	Original uploader was Allefant ¹⁹⁸⁹ at en.wikibooks ¹⁹⁹⁰	

1957 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1958 <http://en.wikibooks.org>

1959 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1960 <http://en.wikibooks.org>

1961 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1962 <http://en.wikibooks.org>

1963 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1964 <http://en.wikibooks.org>

1965 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1966 <http://en.wikibooks.org>

1967 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1968 <http://en.wikibooks.org>

1969 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1970 <http://en.wikibooks.org>

1971 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1972 <http://en.wikibooks.org>

1973 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1974 <http://en.wikibooks.org>

1975 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1976 <http://en.wikibooks.org>

1977 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1978 <http://en.wikibooks.org>

1979 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1980 <http://en.wikibooks.org>

1981 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1982 <http://en.wikibooks.org>

1983 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1984 <http://en.wikibooks.org>

1985 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

1986 <http://en.wikibooks.org>

1987 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>

1988 <http://en.wikibooks.org>

1989 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>

1990 <http://en.wikibooks.org>

756	Original uploader was Allefant ¹⁹⁹¹ at en.wikibooks ¹⁹⁹²	
757	Original uploader was Allefant ¹⁹⁹³ at en.wikibooks ¹⁹⁹⁴	
758	Original uploader was Allefant ¹⁹⁹⁵ at en.wikibooks ¹⁹⁹⁶	
759	Original uploader was Allefant ¹⁹⁹⁷ at en.wikibooks ¹⁹⁹⁸	
760	Original uploader was Allefant ¹⁹⁹⁹ at en.wikibooks ²⁰⁰⁰	
761	Original uploader was Allefant ²⁰⁰¹ at en.wikibooks ²⁰⁰²	
762	Othman Ahmad	GPL
763	Tictoon ²⁰⁰³	GFDL
764	Tictoon ²⁰⁰⁴	GPL
765	Tictoon ²⁰⁰⁵	GPL
766	Tictoon ²⁰⁰⁶	GPL
767	Tictoon ²⁰⁰⁷	GPL
768	Tictoon ²⁰⁰⁸	GPL
769	Original uploader was TBOL3 ²⁰⁰⁹ at en.wikibooks ²⁰¹⁰	
770	Original uploader was TBOL3 ²⁰¹¹ at en.wikibooks ²⁰¹²	
771	Original uploader was AndyD ²⁰¹³ at en.wikibooks ²⁰¹⁴	
772	Original uploader was AndyD ²⁰¹⁵ at en.wikibooks ²⁰¹⁶	
773	Original uploader was AndyD ²⁰¹⁷ at en.wikibooks ²⁰¹⁸	
774	Original uploader was AndyD ²⁰¹⁹ at en.wikibooks ²⁰²⁰	
775	Original uploader was AndyD ²⁰²¹ at en.wikibooks ²⁰²²	

1991 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
 1992 <http://en.wikibooks.org>
 1993 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
 1994 <http://en.wikibooks.org>
 1995 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
 1996 <http://en.wikibooks.org>
 1997 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
 1998 <http://en.wikibooks.org>
 1999 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
 2000 <http://en.wikibooks.org>
 2001 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
 2002 <http://en.wikibooks.org>
 2003 <http://en.wikibooks.org/wiki/User%3ATictoon>
 2004 <http://en.wikibooks.org/wiki/Tictoon>
 2005 <http://en.wikibooks.org/wiki/User%3ATictoon>
 2006 <http://en.wikibooks.org/wiki/User%3ATictoon>
 2007 <http://en.wikibooks.org/wiki/User%3ATictoon>
 2008 <http://en.wikibooks.org/wiki/User%3ATictoon>
 2009 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATBOL3>
 2010 <http://en.wikibooks.org>
 2011 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATBOL3>
 2012 <http://en.wikibooks.org>
 2013 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>
 2014 <http://en.wikibooks.org>
 2015 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>
 2016 <http://en.wikibooks.org>
 2017 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>
 2018 <http://en.wikibooks.org>
 2019 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>
 2020 <http://en.wikibooks.org>
 2021 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>
 2022 <http://en.wikibooks.org>

776	Original uploader was AndyD ²⁰²³ at en.wikibooks ²⁰²⁴	
777	Original uploader was AndyD ²⁰²⁵ at en.wikibooks ²⁰²⁶	
778	Original uploader was AndyD ²⁰²⁷ at en.wikibooks ²⁰²⁸	
779	Original uploader was AndyD ²⁰²⁹ at en.wikibooks ²⁰³⁰	
780	Original uploader was AndyD ²⁰³¹ at en.wikibooks ²⁰³²	
781	Original uploader was AndyD ²⁰³³ at en.wikibooks ²⁰³⁴	
782	Original uploader was AndyD ²⁰³⁵ at en.wikibooks ²⁰³⁶	
783	Original uploader was AndyD ²⁰³⁷ at en.wikibooks ²⁰³⁸	
784	Original uploader was AndyD ²⁰³⁹ at en.wikibooks ²⁰⁴⁰	
785	Original uploader was AndyD ²⁰⁴¹ at en.wikibooks ²⁰⁴²	
786	Original uploader was AndyD ²⁰⁴³ at en.wikibooks ²⁰⁴⁴	
787	Original uploader was AndyD ²⁰⁴⁵ at en.wikibooks ²⁰⁴⁶	
788		cc-by-sa-2.5
789	Jackson Williams (Malaz) ²⁰⁴⁷	PD
790	Blender Foundation	GPL
791	Blender Foundation	GPL
792	Blender Foundation	GPL
793	Blender Foundation	GPL
794	Blender Foundation	GPL
795	Original uploader was Radialronnie ²⁰⁴⁸ at en.wikibooks ²⁰⁴⁹	
796		GFDL
797		GFDL
798	Original uploader was Radialronnie ²⁰⁵⁰ at en.wikibooks ²⁰⁵¹	

²⁰²³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰²⁴ <http://en.wikibooks.org>

²⁰²⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰²⁶ <http://en.wikibooks.org>

²⁰²⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰²⁸ <http://en.wikibooks.org>

²⁰²⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰³⁰ <http://en.wikibooks.org>

²⁰³¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰³² <http://en.wikibooks.org>

²⁰³³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰³⁴ <http://en.wikibooks.org>

²⁰³⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰³⁶ <http://en.wikibooks.org>

²⁰³⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰³⁸ <http://en.wikibooks.org>

²⁰³⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰⁴⁰ <http://en.wikibooks.org>

²⁰⁴¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰⁴² <http://en.wikibooks.org>

²⁰⁴³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰⁴⁴ <http://en.wikibooks.org>

²⁰⁴⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

²⁰⁴⁶ <http://en.wikibooks.org>

²⁰⁴⁷ <http://en.wikibooks.org/wiki/User%3AMalaz>

²⁰⁴⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

²⁰⁴⁹ <http://en.wikibooks.org>

²⁰⁵⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>

²⁰⁵¹ <http://en.wikibooks.org>

799	Original uploader was Radialronnie ²⁰⁵² at en.wikibooks ²⁰⁵³	
800	Original uploader was Radialronnie ²⁰⁵⁴ at en.wikibooks ²⁰⁵⁵	
801	Original uploader was Radialronnie ²⁰⁵⁶ at en.wikibooks ²⁰⁵⁷	
802	Original uploader was Radialronnie ²⁰⁵⁸ at en.wikibooks ²⁰⁵⁹	
803	Original uploader was Radialronnie ²⁰⁶⁰ at en.wikibooks ²⁰⁶¹	
804	SoylentGreen ²⁰⁶²	GPL
805	SoylentGreen ²⁰⁶³	GPL
806	SoylentGreen ²⁰⁶⁴	GPL
807	SoylentGreen ²⁰⁶⁵	GPL
808	SoylentGreen ²⁰⁶⁶	PD
809	SoylentGreen ²⁰⁶⁷	GFDL
810	SoylentGreen ²⁰⁶⁸	GFDL
811	SoylentGreen ²⁰⁶⁹	GPL
812	SoylentGreen ²⁰⁷⁰	GFDL
813	SoylentGreen ²⁰⁷¹	GPL
814	Eilexe ²⁰⁷²	GPL
815	SoylentGreen ²⁰⁷³	GPL
816	SoylentGreen ²⁰⁷⁴	GFDL
817	SoylentGreen ²⁰⁷⁵	GPL
818	SoylentGreen ²⁰⁷⁶	GPL
819	SoylentGreen ²⁰⁷⁷	PD
820	SoylentGreen ²⁰⁷⁸	GPL
821	SoylentGreen ²⁰⁷⁹	cc-by-sa-2.5

2052 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>
 2053 <http://en.wikibooks.org>
 2054 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>
 2055 <http://en.wikibooks.org>
 2056 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>
 2057 <http://en.wikibooks.org>
 2058 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>
 2059 <http://en.wikibooks.org>
 2060 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARadialronnie>
 2061 <http://en.wikibooks.org>
 2062 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2063 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2064 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2065 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2066 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2067 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2068 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2069 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2070 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2071 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2072 <http://de.wikibooks.org/wiki/User%3AEilexe>
 2073 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2074 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2075 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2076 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2077 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2078 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2079 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

822	SoylentGreen ²⁰⁸⁰	GFDL
823	SoylentGreen ²⁰⁸¹	GPL
824	Blender Foundation	GPL
825	SoylentGreen ²⁰⁸²	GPL
826	SoylentGreen ²⁰⁸³	GPL
827	SoylentGreen ²⁰⁸⁴	GFDL
828	SoylentGreen ²⁰⁸⁵	GFDL
829	SoylentGreen ²⁰⁸⁶	GFDL
830	SoylentGreen ²⁰⁸⁷	
831	SoylentGreen ²⁰⁸⁸	GPL
832	SoylentGreen ²⁰⁸⁹	GPL
833	SoylentGreen ²⁰⁹⁰	GPL
834	SoylentGreen ²⁰⁹¹	GFDL
835	SoylentGreen ²⁰⁹²	GFDL
836	SoylentGreen ²⁰⁹³	GPL
837	SoylentGreen ²⁰⁹⁴	GPL
838	SoylentGreen ²⁰⁹⁵	GPL
839	SoylentGreen ²⁰⁹⁶	GPL
840	SoylentGreen ²⁰⁹⁷	GPL
841	SoylentGreen ²⁰⁹⁸	GPL
842	SoylentGreen ²⁰⁹⁹	GPL
843	SoylentGreen ²¹⁰⁰	GPL
844	SoylentGreen ²¹⁰¹	PD
845	SoylentGreen ²¹⁰²	GPL
846	SoylentGreen ²¹⁰³	GPL
847	SoylentGreen ²¹⁰⁴	GPL

2080 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2081 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2082 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2083 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2084 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2085 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2086 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2087 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2088 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2089 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2090 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2091 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2092 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2093 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2094 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2095 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2096 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2097 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2098 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2099 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2100 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2101 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2102 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2103 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2104 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

848	SoylentGreen ²¹⁰⁵	PD
849	SoylentGreen ²¹⁰⁶	PD
850	Original uploader was Porlando ²¹⁰⁷ at en.wikibooks ²¹⁰⁸ . Later version(s) were uploaded by Adrignola ²¹⁰⁹ at en.wikibooks ²¹¹⁰ .	
851	Original uploader was Porlando ²¹¹¹ at en.wikibooks ²¹¹² . Later version(s) were uploaded by Adrignola ²¹¹³ at en.wikibooks ²¹¹⁴ .	
852	Original uploader was Porlando ²¹¹⁵ at en.wikibooks ²¹¹⁶ . Later version(s) were uploaded by Adrignola ²¹¹⁷ at en.wikibooks ²¹¹⁸ .	
853		PD
854	Ldo ²¹¹⁹	cc-by-sa-3.0
855		PD
856		PD
857		PD
858		PD
859		PD
860		PD
861		PD
862		PD
863		PD
864		PD
865		PD
866		PD
867		PD
868		PD
869		PD
870		PD
0	Loremus Ipsemus	None
872	Othman Ahmad ²¹²⁰	cc-by-sa-3.0
873	Original uploader was LokiClock ²¹²¹ at en.wikibooks ²¹²²	GFDL
874	Original uploader was LokiClock ²¹²³ at en.wikibooks ²¹²⁴	

2105 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2106 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2107 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3APorlando>
 2108 <http://en.wikibooks.org>
 2109 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 2110 <http://en.wikibooks.org>
 2111 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3APorlando>
 2112 <http://en.wikibooks.org>
 2113 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 2114 <http://en.wikibooks.org>
 2115 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3APorlando>
 2116 <http://en.wikibooks.org>
 2117 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 2118 <http://en.wikibooks.org>
 2119 <http://en.wikibooks.org/wiki/User%3ALdo>
 2120 <http://en.wikibooks.org/wiki/User%3AOthmanskn2>
 2121 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALokiClock>
 2122 <http://en.wikibooks.org>
 2123 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALokiClock>
 2124 <http://en.wikibooks.org>

875	Original uploader was LokiClock ²¹²⁵ at en.wikibooks ²¹²⁶	
876		PD
877	Jackson Williams (Malaz) ²¹²⁷	PD
878	Jackson Williams (Malaz) ²¹²⁸	PD
879		PD
880	Jackson Williams (Malaz) ²¹²⁹ ; Original uploader was Malaz ²¹³⁰ at en.wikibooks ²¹³¹	PD
881		PD
882	Jackson Williams (Malaz) ²¹³² ; Original uploader was Malaz ²¹³³ at en.wikibooks ²¹³⁴	PD
883	Jackson Williams (Malaz) ²¹³⁵	PD
884	Jackson Williams (Malaz) ²¹³⁶ . Original uploader was Malaz ²¹³⁷ at en.wikibooks ²¹³⁸	PD
885	Jackson Williams (Malaz) ²¹³⁹	PD
886	Jackson Williams (Malaz) ²¹⁴⁰	PD
887	Jackson Williams (Malaz) ²¹⁴¹ . Original uploader was Malaz ²¹⁴² at en.wikibooks ²¹⁴³	PD
888	Jackson Williams (Malaz) ²¹⁴⁴ . Original uploader was Malaz ²¹⁴⁵ at en.wikibooks ²¹⁴⁶	PD
889	Jackson Williams (Malaz) ²¹⁴⁷	PD
890	Original uploader was Greensweater ²¹⁴⁸ at en.wikibooks ²¹⁴⁹	PD
891		PD

2125 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALokiClock>

2126 <http://en.wikibooks.org>

2127 <http://en.wikibooks.org/wiki/User%3AMalaz>

2128 <http://en.wikibooks.org/wiki/User%3AMalaz>

2129 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2130 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2131 <http://en.wikibooks.org>

2132 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2133 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2134 <http://en.wikibooks.org>

2135 <http://en.wikibooks.org/wiki/User%3AMalaz>

2136 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2137 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2138 <http://en.wikibooks.org>

2139 <http://en.wikibooks.org/wiki/User%3AMalaz>

2140 <http://en.wikibooks.org/wiki/User%3AMalaz>

2141 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2142 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2143 <http://en.wikibooks.org>

2144 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2145 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMalaz>

2146 <http://en.wikibooks.org>

2147 <http://en.wikibooks.org/wiki/User%3AMalaz>

2148 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>

2149 <http://en.wikibooks.org>

892	Original uploader was Greensweater ²¹⁵⁰ at en.wikibooks ²¹⁵¹ . Later version(s) were uploaded by Adrignola ²¹⁵² at en.wikibooks ²¹⁵³ .	
893	Original uploader was Greensweater ²¹⁵⁴ at en.wikibooks ²¹⁵⁵ . Later version(s) were uploaded by Adrignola ²¹⁵⁶ at en.wikibooks ²¹⁵⁷ .	
894	Original uploader was Greensweater ²¹⁵⁸ at en.wikibooks ²¹⁵⁹	
895	Original uploader was Greensweater ²¹⁶⁰ at en.wikibooks ²¹⁶¹ . Later version(s) were uploaded by Adrignola ²¹⁶² at en.wikibooks ²¹⁶³ .	
896	Original uploader was Greensweater ²¹⁶⁴ at en.wikibooks ²¹⁶⁵	
897	Original uploader was Greensweater ²¹⁶⁶ at en.wikibooks ²¹⁶⁷	
898	Original uploader was Greensweater ²¹⁶⁸ at en.wikibooks ²¹⁶⁹ . Later version(s) were uploaded by Adrignola ²¹⁷⁰ at en.wikibooks ²¹⁷¹ .	
899	Original uploader was Greensweater ²¹⁷² at en.wikibooks ²¹⁷³	
900	Original uploader was Greensweater ²¹⁷⁴ at en.wikibooks ²¹⁷⁵	
901	Original uploader was Greensweater ²¹⁷⁶ at en.wikibooks ²¹⁷⁷ . Later version(s) were uploaded by Adrignola ²¹⁷⁸ at en.wikibooks ²¹⁷⁹ .	

-
- 2150 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2151 <http://en.wikibooks.org>
 - 2152 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 - 2153 <http://en.wikibooks.org>
 - 2154 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2155 <http://en.wikibooks.org>
 - 2156 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 - 2157 <http://en.wikibooks.org>
 - 2158 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2159 <http://en.wikibooks.org>
 - 2160 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2161 <http://en.wikibooks.org>
 - 2162 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 - 2163 <http://en.wikibooks.org>
 - 2164 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2165 <http://en.wikibooks.org>
 - 2166 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2167 <http://en.wikibooks.org>
 - 2168 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2169 <http://en.wikibooks.org>
 - 2170 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 - 2171 <http://en.wikibooks.org>
 - 2172 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2173 <http://en.wikibooks.org>
 - 2174 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2175 <http://en.wikibooks.org>
 - 2176 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
 - 2177 <http://en.wikibooks.org>
 - 2178 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
 - 2179 <http://en.wikibooks.org>

902	Original uploader was Greensweater ²¹⁸⁰ at en.wikibooks ²¹⁸¹ . Later version(s) were uploaded by Adrignola ²¹⁸² at en.wikibooks ²¹⁸³ .	
903	Original uploader was Greensweater ²¹⁸⁴ at en.wikibooks ²¹⁸⁵	
904	Original uploader was Greensweater ²¹⁸⁶ at en.wikibooks ²¹⁸⁷	PD
905		PD
0	Loremus Ipsemus	None
907	Lawrence D'Oliveiro	GPL
908	Lawrence D'Oliveiro	GPL
909	Lawrence D'Oliveiro	GPL
910	Lawrence D'Oliveiro	GPL
911	Lawrence D'Oliveiro	GPL
912	Lawrence D'Oliveiro	GPL
913		GPL
914		GPL
915	Kesten ²¹⁸⁸	PD
916	Wayne Harrison	cc-by-sa-3.0
917		PD
918		PD
919		PD
920		PD
921		PD
922		GFDL
923	Original uploader was Aidan.Sullivan ²¹⁸⁹ at en.wikibooks ²¹⁹⁰ . Later version(s) were uploaded by Adrignola ²¹⁹¹ , Swift ²¹⁹² , Dal308 ²¹⁹³ at en.wikibooks ²¹⁹⁴ .	
924	Original uploader was Ascen ²¹⁹⁵ at en.wikibooks ²¹⁹⁶	GPL
925	Original uploader was Ascen ²¹⁹⁷ at en.wikibooks ²¹⁹⁸	GPL
926	Original uploader was Aidan.Sullivan ²¹⁹⁹ at en.wikibooks ²²⁰⁰	GPL

2180 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>

2181 <http://en.wikibooks.org>

2182 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2183 <http://en.wikibooks.org>

2184 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>

2185 <http://en.wikibooks.org>

2186 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>

2187 <http://en.wikibooks.org>

2188 <http://en.wikibooks.org/wiki/User%3AKesten>

2189 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAidan.Sullivan>

2190 <http://en.wikibooks.org>

2191 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2192 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASwift>

2193 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADal308>

2194 <http://en.wikibooks.org>

2195 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAscen>

2196 <http://en.wikibooks.org>

2197 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAscen>

2198 <http://en.wikibooks.org>

2199 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAidan.Sullivan>

2200 <http://en.wikibooks.org>

927	Original uploader was Aidan.Sullivan ²²⁰¹ at en.wikibooks ²²⁰²	GPL
928		GFDL
929		GFDL
930		GFDL
931		GFDL
932	. Original uploader was Moohasha ²²⁰³ at en.wikibooks ²²⁰⁴	PD
933	Moohasha ²²⁰⁵ . Original uploader was Moohasha ²²⁰⁶ at en.wikibooks ²²⁰⁷	
934	Original uploader was Moohasha ²²⁰⁸ at en.wikibooks ²²⁰⁹	
935	Moohasha ²²¹⁰ . Original uploader was Moohasha ²²¹¹ at en.wikibooks ²²¹²	
936	Moohasha ²²¹³ . Original uploader was Moohasha ²²¹⁴ at en.wikibooks ²²¹⁵	
937	Original uploader was Moohasha ²²¹⁶ at en.wikibooks ²²¹⁷	
938	Moohasha ²²¹⁸ . Original uploader was Moohasha ²²¹⁹ at en.wikibooks ²²²⁰	
939	Original uploader was Moohasha ²²²¹ at en.wikibooks ²²²²	
940	Moohasha ²²²³ . Original uploader was Moohasha ²²²⁴ at en.wikibooks ²²²⁵	
941		GFDL
942	Original uploader was Moohasha ²²²⁶ at en.wikibooks ²²²⁷	
943	Moohasha ²²²⁸ . Original uploader was Moohasha ²²²⁹ at en.wikibooks ²²³⁰	

2201 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAidan.Sullivan>

2202 <http://en.wikibooks.org>

2203 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2204 <http://en.wikibooks.org>

2205 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2206 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2207 <http://en.wikibooks.org>

2208 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2209 <http://en.wikibooks.org>

2210 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2211 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2212 <http://en.wikibooks.org>

2213 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2214 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2215 <http://en.wikibooks.org>

2216 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2217 <http://en.wikibooks.org>

2218 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2219 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2220 <http://en.wikibooks.org>

2221 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2222 <http://en.wikibooks.org>

2223 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2224 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2225 <http://en.wikibooks.org>

2226 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2227 <http://en.wikibooks.org>

2228 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2229 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2230 <http://en.wikibooks.org>

944	Moohasha ²²³¹ . en.wikibooks ²²³³	Original uploader was Moohasha ²²³² at	
945		Original uploader was Moohasha ²²³⁴ at en.wikibooks ²²³⁵	
946	Moohasha ²²³⁶		PD
947	Moohasha ²²³⁷ . en.wikibooks ²²³⁹	Original uploader was Moohasha ²²³⁸ at	
948	Moohasha ²²⁴⁰ . en.wikibooks ²²⁴²	Original uploader was Moohasha ²²⁴¹ at	
949	Moohasha ²²⁴³ . en.wikibooks ²²⁴⁵	Original uploader was Moohasha ²²⁴⁴ at	
950	Moohasha ²²⁴⁶ . en.wikibooks ²²⁴⁸	Original uploader was Moohasha ²²⁴⁷ at	
951	Moohasha ²²⁴⁹ . en.wikibooks ²²⁵¹	Original uploader was Moohasha ²²⁵⁰ at	GFDL
952	Moohasha ²²⁵² . en.wikibooks ²²⁵⁴	Original uploader was Moohasha ²²⁵³ at	GFDL
953	Moohasha ²²⁵⁵ . en.wikibooks ²²⁵⁷	Original uploader was Moohasha ²²⁵⁶ at	GFDL
954	Moohasha ²²⁵⁸ . en.wikibooks ²²⁶⁰	Original uploader was Moohasha ²²⁵⁹ at	GFDL

2231 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2232 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2233 <http://en.wikibooks.org>

2234 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2235 <http://en.wikibooks.org>

2236 <http://en.wikibooks.org/wiki/User%3AMoohasha>

2237 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2238 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2239 <http://en.wikibooks.org>

2240 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2241 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2242 <http://en.wikibooks.org>

2243 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2244 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2245 <http://en.wikibooks.org>

2246 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2247 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2248 <http://en.wikibooks.org>

2249 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2250 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2251 <http://en.wikibooks.org>

2252 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2253 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2254 <http://en.wikibooks.org>

2255 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2256 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2257 <http://en.wikibooks.org>

2258 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2259 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>

2260 <http://en.wikibooks.org>

955	Moohasha ²²⁶¹ . en.wikibooks ²²⁶³	Original uploader was Moohasha ²²⁶² at	GFDL
956	Moohasha ²²⁶⁴ . en.wikibooks ²²⁶⁶	Original uploader was Moohasha ²²⁶⁵ at	GFDL
957	Moohasha ²²⁶⁷ . en.wikibooks ²²⁶⁹	Original uploader was Moohasha ²²⁶⁸ at	GFDL
958	Moohasha ²²⁷⁰ . en.wikibooks ²²⁷²	Original uploader was Moohasha ²²⁷¹ at	GFDL
959			GFDL
960			GFDL
961			GFDL
962			GFDL
963			GFDL
964			GFDL
965			GFDL
966			GFDL
967			GFDL
968	SoylentGreen ²²⁷³		PD
969	SoylentGreen ²²⁷⁴		GPL
970	SoylentGreen ²²⁷⁵		PD
971	SoylentGreen ²²⁷⁶		GPL
972	SoylentGreen ²²⁷⁷		GPL
973	SoylentGreen ²²⁷⁸		PD
974	Jeremy B.. en.wikibooks ²²⁸⁰	Original uploader was Bullercruz1 ²²⁷⁹ at	
975	Jeremy B.. en.wikibooks ²²⁸²	Original uploader was Bullercruz1 ²²⁸¹ at	

2261 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2262 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2263 <http://en.wikibooks.org>
 2264 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2265 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2266 <http://en.wikibooks.org>
 2267 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2268 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2269 <http://en.wikibooks.org>
 2270 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2271 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMoohasha>
 2272 <http://en.wikibooks.org>
 2273 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2274 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2275 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2276 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2277 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2278 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2279 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>
 2280 <http://en.wikibooks.org>
 2281 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>
 2282 <http://en.wikibooks.org>

976	Jeremy B.. Original uploader was Bullercruz1 ²²⁸³ at en.wikibooks ²²⁸⁴	
977	Jeremy B. Original uploader was Bullercruz1 ²²⁸⁵ at en.wikibooks ²²⁸⁶ . Later version(s) were uploaded by Adrignola ²²⁸⁷ at en.wikibooks ²²⁸⁸ .	PD
978	Jeremy B.	PD
979	Jeremy B.	PD
980	Jeremy B.	PD
981	Jeremy B.	PD
982	Jeremy B.	PD
983	Jeremy B. Original uploader was Bullercruz1 ²²⁸⁹ at en.wikibooks ²²⁹⁰ . Later version(s) were uploaded by Adrignola ²²⁹¹ at en.wikibooks ²²⁹² .	PD
984	Jeremy B. Original uploader was Bullercruz1 ²²⁹³ at en.wikibooks ²²⁹⁴ . Later version(s) were uploaded by Adrignola ²²⁹⁵ at en.wikibooks ²²⁹⁶ .	PD
985	Jeremy B.. Original uploader was Bullercruz1 ²²⁹⁷ at en.wikibooks ²²⁹⁸	
986	Jeremy B.	PD
987	Jeremy B.. Original uploader was Bullercruz1 ²²⁹⁹ at en.wikibooks ²³⁰⁰ . Later version(s) were uploaded by Adrignola ²³⁰¹ at en.wikibooks ²³⁰² .	PD
988	Lawrence D'Oliveiro	GPL
989	Lawrence D'Oliveiro	GPL
990	Lawrence D'Oliveiro	GPL
991	Lawrence D'Oliveiro	GPL
992	Lawrence D'Oliveiro	GPL
993	Lawrence D'Oliveiro	GPL
994	Original uploader was Gabio ²³⁰³ at en.wikibooks ²³⁰⁴	
995	Nkansah Rexford	cc-by-sa-3.0

2283 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2284 <http://en.wikibooks.org>

2285 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2286 <http://en.wikibooks.org>

2287 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2288 <http://en.wikibooks.org>

2289 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2290 <http://en.wikibooks.org>

2291 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2292 <http://en.wikibooks.org>

2293 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2294 <http://en.wikibooks.org>

2295 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2296 <http://en.wikibooks.org>

2297 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2298 <http://en.wikibooks.org>

2299 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2300 <http://en.wikibooks.org>

2301 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2302 <http://en.wikibooks.org>

2303 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2304 <http://en.wikibooks.org>

996	Nkansah Rexford	cc-by-sa-3.0
997	Original uploader was Gabio ²³⁰⁵ at en.wikibooks ²³⁰⁶	
998	Original uploader was Gabio ²³⁰⁷ at en.wikibooks ²³⁰⁸	
999	Original uploader was Gabio ²³⁰⁹ at en.wikibooks ²³¹⁰	PD
1000	Original uploader was Gabio ²³¹¹ at en.wikibooks ²³¹²	PD
1001	Original uploader was Gabio ²³¹³ at en.wikibooks ²³¹⁴	PD
1002	Original uploader was Gabio ²³¹⁵ at en.wikibooks ²³¹⁶	PD
1003	Original uploader was Gabio ²³¹⁷ at en.wikibooks ²³¹⁸	PD
1004	Original uploader was Gabio ²³¹⁹ at en.wikibooks ²³²⁰	PD
1005	Original uploader was Gabio ²³²¹ at en.wikibooks ²³²²	PD
1006	Original uploader was Gabio ²³²³ at en.wikibooks ²³²⁴	
1007	Original uploader was Gabio ²³²⁵ at en.wikibooks ²³²⁶	PD
1008	Original uploader was Gabio ²³²⁷ at en.wikibooks ²³²⁸	PD
1009	Original uploader was Gabio ²³²⁹ at en.wikibooks ²³³⁰	PD
1010	Original uploader was Gabio ²³³¹ at en.wikibooks ²³³²	PD
1011	Original uploader was Gabio ²³³³ at en.wikibooks ²³³⁴	PD
1012	Original uploader was Gabio ²³³⁵ at en.wikibooks ²³³⁶	PD
1013	Original uploader was Gabio ²³³⁷ at en.wikibooks ²³³⁸	PD

2305 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2306 <http://en.wikibooks.org>

2307 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2308 <http://en.wikibooks.org>

2309 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2310 <http://en.wikibooks.org>

2311 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2312 <http://en.wikibooks.org>

2313 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2314 <http://en.wikibooks.org>

2315 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2316 <http://en.wikibooks.org>

2317 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2318 <http://en.wikibooks.org>

2319 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2320 <http://en.wikibooks.org>

2321 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2322 <http://en.wikibooks.org>

2323 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2324 <http://en.wikibooks.org>

2325 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2326 <http://en.wikibooks.org>

2327 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2328 <http://en.wikibooks.org>

2329 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2330 <http://en.wikibooks.org>

2331 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2332 <http://en.wikibooks.org>

2333 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2334 <http://en.wikibooks.org>

2335 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2336 <http://en.wikibooks.org>

2337 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2338 <http://en.wikibooks.org>

1014	Original uploader was Gabio ²³³⁹ at en.wikibooks ²³⁴⁰	
1015	Original uploader was Gabio ²³⁴¹ at en.wikibooks ²³⁴²	PD
1016	Original uploader was Gabio ²³⁴³ at en.wikibooks ²³⁴⁴	PD
1017	Original uploader was Gabio ²³⁴⁵ at en.wikibooks ²³⁴⁶	PD
1018	Original uploader was Gabio ²³⁴⁷ at en.wikibooks ²³⁴⁸	
1019	Original uploader was Gabio ²³⁴⁹ at en.wikibooks ²³⁵⁰	PD
1020	Original uploader was Gabio ²³⁵¹ at en.wikibooks ²³⁵²	PD
1021	Original uploader was Gabio ²³⁵³ at en.wikibooks ²³⁵⁴	PD
1022	Original uploader was Gabio ²³⁵⁵ at en.wikibooks ²³⁵⁶	PD
1023	Original uploader was Gabio ²³⁵⁷ at en.wikibooks ²³⁵⁸	
1024	Original uploader was Gabio ²³⁵⁹ at en.wikibooks ²³⁶⁰	
1025	Original uploader was Gabio ²³⁶¹ at en.wikibooks ²³⁶²	PD
1026	Original uploader was Gabio ²³⁶³ at en.wikibooks ²³⁶⁴	
1027	Gabio	PD
1028	Original uploader was Gabio ²³⁶⁵ at en.wikibooks ²³⁶⁶	
1029	Original uploader was Gabio ²³⁶⁷ at en.wikibooks ²³⁶⁸	PD
1030	Original uploader was Stalepanda ²³⁶⁹ at en.wikibooks ²³⁷⁰	GFDL
1031	Original uploader was Stalepanda ²³⁷¹ at en.wikibooks ²³⁷²	GFDL

²³³⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁴⁰ <http://en.wikibooks.org>

²³⁴¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁴² <http://en.wikibooks.org>

²³⁴³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁴⁴ <http://en.wikibooks.org>

²³⁴⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁴⁶ <http://en.wikibooks.org>

²³⁴⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁴⁸ <http://en.wikibooks.org>

²³⁴⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁵⁰ <http://en.wikibooks.org>

²³⁵¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁵² <http://en.wikibooks.org>

²³⁵³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁵⁴ <http://en.wikibooks.org>

²³⁵⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁵⁶ <http://en.wikibooks.org>

²³⁵⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁵⁸ <http://en.wikibooks.org>

²³⁵⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁶⁰ <http://en.wikibooks.org>

²³⁶¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁶² <http://en.wikibooks.org>

²³⁶³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁶⁴ <http://en.wikibooks.org>

²³⁶⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁶⁶ <http://en.wikibooks.org>

²³⁶⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²³⁶⁸ <http://en.wikibooks.org>

²³⁶⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AStalepanda>

²³⁷⁰ <http://en.wikibooks.org>

²³⁷¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AStalepanda>

²³⁷² <http://en.wikibooks.org>

1032	Original uploader was Stalepanda ²³⁷³ at en.wikibooks ²³⁷⁴	GFDL
1033	Original uploader was Stalepanda ²³⁷⁵ at en.wikibooks ²³⁷⁶	GFDL
1034	Original uploader was Gabio ²³⁷⁷ at en.wikibooks ²³⁷⁸	PD
1035		PD
1036		PD
1037	Original uploader was Gabio ²³⁷⁹ at en.wikibooks ²³⁸⁰	
1038		GPL
1039		cc-by-sa-2.5
1040		cc-by-sa-2.5
1041		cc-by-sa-2.5
1042		GPL
1043	Original uploader was AndyD ²³⁸¹ at en.wikibooks ²³⁸²	
1044		GPL
1045	Original uploader was AndyD ²³⁸³ at en.wikibooks ²³⁸⁴	
1046		GPL
1047		GPL
1048	Original uploader was Gabio ²³⁸⁵ at en.wikibooks ²³⁸⁶	
1049	Original uploader was Gabio ²³⁸⁷ at en.wikibooks ²³⁸⁸	PD
1050	Original uploader was Gabio ²³⁸⁹ at en.wikibooks ²³⁹⁰	
1051	Original uploader was Gabio ²³⁹¹ at en.wikibooks ²³⁹²	PD
1052	Original uploader was Gabio ²³⁹³ at en.wikibooks ²³⁹⁴	PD
1053	Original uploader was Gabio ²³⁹⁵ at en.wikibooks ²³⁹⁶	
1054	Original uploader was Gabio ²³⁹⁷ at en.wikibooks ²³⁹⁸	PD
1055	Original uploader was Gabio ²³⁹⁹ at en.wikibooks ²⁴⁰⁰	

2373 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AStalepanda>

2374 <http://en.wikibooks.org>

2375 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AStalepanda>

2376 <http://en.wikibooks.org>

2377 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2378 <http://en.wikibooks.org>

2379 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2380 <http://en.wikibooks.org>

2381 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

2382 <http://en.wikibooks.org>

2383 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAndyD>

2384 <http://en.wikibooks.org>

2385 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2386 <http://en.wikibooks.org>

2387 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2388 <http://en.wikibooks.org>

2389 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2390 <http://en.wikibooks.org>

2391 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2392 <http://en.wikibooks.org>

2393 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2394 <http://en.wikibooks.org>

2395 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2396 <http://en.wikibooks.org>

2397 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2398 <http://en.wikibooks.org>

2399 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2400 <http://en.wikibooks.org>

1056	Original uploader was Gabio ²⁴⁰¹ at en.wikibooks ²⁴⁰²	PD
1057	Original uploader was Gabio ²⁴⁰³ at en.wikibooks ²⁴⁰⁴	PD
1058		GFDL
1059	Original uploader was Gabio ²⁴⁰⁵ at en.wikibooks ²⁴⁰⁶	PD
1060	Original uploader was Gabio ²⁴⁰⁷ at en.wikibooks ²⁴⁰⁸	
1061	Original uploader was Gabio ²⁴⁰⁹ at en.wikibooks ²⁴¹⁰	PD
1062	Original uploader was Gabio ²⁴¹¹ at en.wikibooks ²⁴¹²	PD
1063	Original uploader was Gabio ²⁴¹³ at en.wikibooks ²⁴¹⁴	PD
1064	Original uploader was Gabio ²⁴¹⁵ at en.wikibooks ²⁴¹⁶	PD
1065	Original uploader was Gabio ²⁴¹⁷ at en.wikibooks ²⁴¹⁸	PD
1066	Original uploader was Gabio ²⁴¹⁹ at en.wikibooks ²⁴²⁰	PD
1067	Original uploader was Gabio ²⁴²¹ at en.wikibooks ²⁴²²	
1068	Original uploader was Halley ²⁴²³ at en.wikibooks ²⁴²⁴	GFDL
1069		GFDL
1070	Original uploader was Halley ²⁴²⁵ at en.wikibooks ²⁴²⁶	GFDL
1071		GFDL
1072		GFDL
1073		GFDL
1074		GFDL
1075		GFDL
1076		GFDL
1077		GFDL
1078		GFDL
1079		GFDL
1080		GFDL

²⁴⁰¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴⁰² <http://en.wikibooks.org>

²⁴⁰³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴⁰⁴ <http://en.wikibooks.org>

²⁴⁰⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴⁰⁶ <http://en.wikibooks.org>

²⁴⁰⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴⁰⁸ <http://en.wikibooks.org>

²⁴⁰⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴¹⁰ <http://en.wikibooks.org>

²⁴¹¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴¹² <http://en.wikibooks.org>

²⁴¹³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴¹⁴ <http://en.wikibooks.org>

²⁴¹⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴¹⁶ <http://en.wikibooks.org>

²⁴¹⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴¹⁸ <http://en.wikibooks.org>

²⁴¹⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴²⁰ <http://en.wikibooks.org>

²⁴²¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

²⁴²² <http://en.wikibooks.org>

²⁴²³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHalley>

²⁴²⁴ <http://en.wikibooks.org>

²⁴²⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AHalley>

²⁴²⁶ <http://en.wikibooks.org>

1081	Soylentgreen ²⁴²⁷	GFDL
1082	Soylentgreen ²⁴²⁸	GFDL
1083	Soylentgreen ²⁴²⁹	GFDL
1084	Soylentgreen ²⁴³⁰	GFDL
1085	Soylentgreen ²⁴³¹	GFDL
1086	Soylentgreen ²⁴³²	GFDL
1087	Soylentgreen ²⁴³³	GFDL
1088	Soylentgreen ²⁴³⁴	GFDL
1089	SoylentGreen ²⁴³⁵	GFDL
1090	SoylentGreen ²⁴³⁶	GPL
1091	SoylentGreen ²⁴³⁷	GPL
1092	SoylentGreen ²⁴³⁸	GFDL
1093	SoylentGreen ²⁴³⁹	GPL
1094	SoylentGreen ²⁴⁴⁰	GPL
1095	SoylentGreen ²⁴⁴¹	GPL
1096	SoylentGreen ²⁴⁴²	GPL
1097	SoylentGreen ²⁴⁴³	GPL
1098	SoylentGreen ²⁴⁴⁴	GFDL
1099	SoylentGreen ²⁴⁴⁵	GPL
1100	SoylentGreen ²⁴⁴⁶	GFDL
1101	SoylentGreen ²⁴⁴⁷	GFDL
1102	SoylentGreen ²⁴⁴⁸	PD
1103	SoylentGreen ²⁴⁴⁹	GFDL
1104	SoylentGreen ²⁴⁵⁰	GFDL
1105	Roger McLassus ²⁴⁵¹	GFDL

2427 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2428 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2429 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2430 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2431 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2432 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2433 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2434 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
 2435 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2436 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2437 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2438 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2439 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2440 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2441 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2442 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2443 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2444 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2445 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2446 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2447 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2448 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2449 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2450 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
 2451 <http://en.wikibooks.org/wiki/User%3ARoger%20McLassus>

1106	Jeremy B. Original uploader was Bullercruz1 ²⁴⁵² at en.wikibooks ²⁴⁵³ . Later version(s) were uploaded by Adrignola ²⁴⁵⁴ at en.wikibooks ²⁴⁵⁵ .	PD
1107	Jeremy B. Original uploader was Bullercruz1 ²⁴⁵⁶ at en.wikibooks ²⁴⁵⁷ . Later version(s) were uploaded by Adrignola ²⁴⁵⁸ at en.wikibooks ²⁴⁵⁹ .	PD
1108	Jeremy Buller ²⁴⁶⁰ . Original uploader was Bullercruz1 ²⁴⁶¹ at en.wikibooks ²⁴⁶² . Later version(s) were uploaded by Adrignola ²⁴⁶³ at en.wikibooks ²⁴⁶⁴ .	
1109	Jeremy B. Original uploader was Bullercruz1 ²⁴⁶⁵ at en.wikibooks ²⁴⁶⁶	
1110	Jeremy B. Original uploader was Bullercruz1 ²⁴⁶⁷ at en.wikibooks ²⁴⁶⁸ . Later version(s) were uploaded by Adrignola ²⁴⁶⁹ at en.wikibooks ²⁴⁷⁰ .	PD
1111	Jeremy B.. Original uploader was Bullercruz1 ²⁴⁷¹ at en.wikibooks ²⁴⁷² . Later version(s) were uploaded by Adrignola ²⁴⁷³ at en.wikibooks ²⁴⁷⁴ .	PD
1112	Jeremy B. Original uploader was Bullercruz1 ²⁴⁷⁵ at en.wikibooks ²⁴⁷⁶ . Later version(s) were uploaded by Adrignola ²⁴⁷⁷ at en.wikibooks ²⁴⁷⁸ .	PD

²⁴⁵² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁵³ <http://en.wikibooks.org>

²⁴⁵⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

²⁴⁵⁵ <http://en.wikibooks.org>

²⁴⁵⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁵⁷ <http://en.wikibooks.org>

²⁴⁵⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

²⁴⁵⁹ <http://en.wikibooks.org>

²⁴⁶⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3Abullercruz1>

²⁴⁶¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁶² <http://en.wikibooks.org>

²⁴⁶³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

²⁴⁶⁴ <http://en.wikibooks.org>

²⁴⁶⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁶⁶ <http://en.wikibooks.org>

²⁴⁶⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁶⁸ <http://en.wikibooks.org>

²⁴⁶⁹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

²⁴⁷⁰ <http://en.wikibooks.org>

²⁴⁷¹ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁷² <http://en.wikibooks.org>

²⁴⁷³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

²⁴⁷⁴ <http://en.wikibooks.org>

²⁴⁷⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

²⁴⁷⁶ <http://en.wikibooks.org>

²⁴⁷⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

²⁴⁷⁸ <http://en.wikibooks.org>

1113	Jeremy B. Original uploader was Bullercruz1 ²⁴⁷⁹ at en.wikibooks ²⁴⁸⁰ . Later version(s) were uploaded by Adrignola ²⁴⁸¹ at en.wikibooks ²⁴⁸² .	PD
1114	Jeremy B. Original uploader was Bullercruz1 ²⁴⁸³ at en.wikibooks ²⁴⁸⁴ . Later version(s) were uploaded by Adrignola ²⁴⁸⁵ at en.wikibooks ²⁴⁸⁶ .	PD
1115	Jeremy B.. Original uploader was Bullercruz1 ²⁴⁸⁷ at en.wikibooks ²⁴⁸⁸ . Later version(s) were uploaded by Adrignola ²⁴⁸⁹ at en.wikibooks ²⁴⁹⁰ .	
1116	Jeremy B. Original uploader was Bullercruz1 ²⁴⁹¹ at en.wikibooks ²⁴⁹² . Later version(s) were uploaded by Adrignola ²⁴⁹³ at en.wikibooks ²⁴⁹⁴ .	PD
1117	Guillaume22 ²⁴⁹⁵	GFDL
1118	Original uploader was LuizAlvarez ²⁴⁹⁶ at en.wikibooks ²⁴⁹⁷	
1119	Original uploader was Goeland86 ²⁴⁹⁸ at en.wikibooks ²⁴⁹⁹	
1120		GPL
1121	BenutzernameAufCommons ²⁵⁰⁰	cc-by-2.5
1122	User:Toni Grappa ²⁵⁰¹	
1123	SoylentGreen ²⁵⁰²	GFDL
1124	Original uploader was Jawboot ²⁵⁰³ at en.wikibooks ²⁵⁰⁴	PD
1125	User:Toni Grappa ²⁵⁰⁵	
1126	b:en>User:Siggimund ²⁵⁰⁶	PD
1127	SoylentGreen ²⁵⁰⁷	cc-by-2.5

2479 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2480 <http://en.wikibooks.org>

2481 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2482 <http://en.wikibooks.org>

2483 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2484 <http://en.wikibooks.org>

2485 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2486 <http://en.wikibooks.org>

2487 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2488 <http://en.wikibooks.org>

2489 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2490 <http://en.wikibooks.org>

2491 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABullercruz1>

2492 <http://en.wikibooks.org>

2493 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2494 <http://en.wikibooks.org>

2495 <http://en.wikibooks.org/wiki/User%3AGuillaume22>

2496 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ALuizAlvarez>

2497 <http://en.wikibooks.org>

2498 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGoeland86>

2499 <http://en.wikibooks.org>

2500 <http://en.wikibooks.org/wiki/User%3ABenutzernameAufCommons>

2501 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2502 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2503 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJawboot>

2504 <http://en.wikibooks.org>

2505 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2506 <http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASiggimund>

2507 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

1128	Imageday ²⁵⁰⁸	GFDL
1129	Guillaume22 ²⁵⁰⁹	GFDL
1130	Blender Foundation	GPL
1131	SoylentGreen ²⁵¹⁰	PD
1132	Alex Grace	GFDL
1133	Original uploader was Totophe64 ²⁵¹¹ at en.wikibooks ²⁵¹²	
1134	User:Toni Grappa ²⁵¹³	
1135	Guillaume22 ²⁵¹⁴	GFDL
1136	Original uploader was Gabio ²⁵¹⁵ at en.wikibooks ²⁵¹⁶	
1137	Original uploader was Totophe64 ²⁵¹⁷ at en.wikibooks ²⁵¹⁸	
1138	SoylentGreen ²⁵¹⁹	GFDL
1139	User:Toni Grappa ²⁵²⁰	
1140	Mangarm ²⁵²¹	GPL
1141	Blender Foundation	GPL
1142	Blender Foundation	GPL
1143	Blender Foundation	GPL
1144	Toni Grappa ²⁵²²	GFDL
1145	Original uploader was ZeroOne ²⁵²³ at en.wikibooks ²⁵²⁴	
1146	User:Toni Grappa ²⁵²⁵	
1147	User:Toni Grappa ²⁵²⁶	
1148	Toni Grappa ²⁵²⁷	
1149	Original uploader was Gabio ²⁵²⁸ at en.wikibooks ²⁵²⁹	PD
1150	Original uploader was MikaYuodas ²⁵³⁰ at fr.wikiversity ²⁵³¹	GFDL
1151		

2508 <http://en.wikibooks.org/wiki/User%3AImageday>

2509 <http://en.wikibooks.org/wiki/User%3AGuillaume22>

2510 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2511 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

2512 <http://en.wikibooks.org>

2513 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2514 <http://en.wikibooks.org/wiki/User%3AGuillaume22>

2515 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2516 <http://en.wikibooks.org>

2517 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

2518 <http://en.wikibooks.org>

2519 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2520 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2521 <http://en.wikibooks.org/wiki/User%3AMangarm>

2522 http://en.wikibooks.org/wiki/User%3AToni_grappa

2523 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne>

2524 <http://en.wikibooks.org>

2525 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2526 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2527 http://en.wikibooks.org/wiki/User%3AToni_grappa

2528 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2529 <http://en.wikibooks.org>

2530 <http://en.wikibooks.org/wiki/%3Awikiversity%3Afr%3AUser%3AMikaYuodas>

2531 <http://fr.wikiversity.org>

1152	Original uploader was Sumoncil ²⁵³² at en.wikibooks ²⁵³³ . Later version(s) were uploaded by Adrignola ²⁵³⁴ at en.wikibooks ²⁵³⁵ .	
1153	Original uploader was Sumoncil ²⁵³⁶ at en.wikibooks ²⁵³⁷ . Later version(s) were uploaded by Adrignola ²⁵³⁸ at en.wikibooks ²⁵³⁹ .	
1154	Blender Foundation	GPL
1155	Original uploader was ZeroOne ²⁵⁴⁰ at en.wikibooks ²⁵⁴¹	
1156	Soylentgreen ²⁵⁴²	GFDL
1157	CwadrupldijitGoldilox ²⁵⁴³	PD
1158	Banlu Kemiyaorn ²⁵⁴⁴	PD
1159	Blender Foundation	GPL
1160	User:Toni Grappa ²⁵⁴⁵	
1161	John Whelan. Original uploader was Johnwhelan ²⁵⁴⁶ at en.wikibooks ²⁵⁴⁷ . Later version(s) were uploaded by Adrignola ²⁵⁴⁸ at en.wikibooks ²⁵⁴⁹ .	
1162	Original uploader was Argento ²⁵⁵⁰ at en.wikibooks ²⁵⁵¹	
1163		GFDL
1164	Blender Foundation	GPL
1165	b:en>User:Lazy-lump ²⁵⁵²	PD
1166	Original uploader was Gregggreg ²⁵⁵³ at en.wikibooks ²⁵⁵⁴	
1167	Gabio	PD
1168	Original uploader was Argento ²⁵⁵⁵ at en.wikibooks ²⁵⁵⁶	
1169	SaphireS	cc-by-sa-3.0
1170	b:en>User:Spiderworm ²⁵⁵⁷	PD
1171	Original uploader was AverieZen ²⁵⁵⁸ at en.wikibooks ²⁵⁵⁹	

[2532 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[2533 http://en.wikibooks.org](http://en.wikibooks.org)
[2534 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[2535 http://en.wikibooks.org](http://en.wikibooks.org)
[2536 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil)
[2537 http://en.wikibooks.org](http://en.wikibooks.org)
[2538 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[2539 http://en.wikibooks.org](http://en.wikibooks.org)
[2540 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne)
[2541 http://en.wikibooks.org](http://en.wikibooks.org)
[2542 http://en.wikibooks.org/wiki/User%3ASoylentgreen](http://en.wikibooks.org/wiki/User%3ASoylentgreen)
[2543 http://en.wikibooks.org/wiki/User%3ACwadrupldijitGoldilox](http://en.wikibooks.org/wiki/User%3ACwadrupldijitGoldilox)
[2544 http://en.wikibooks.org/wiki/User%3ABanlu%20Kemiyaorn](http://en.wikibooks.org/wiki/User%3ABanlu%20Kemiyaorn)
[2545 http://en.wikibooks.org/wiki/User%3AToni%20Grappa](http://en.wikibooks.org/wiki/User%3AToni%20Grappa)
[2546 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohnwhelan](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohnwhelan)
[2547 http://en.wikibooks.org](http://en.wikibooks.org)
[2548 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola)
[2549 http://en.wikibooks.org](http://en.wikibooks.org)
[2550 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento)
[2551 http://en.wikibooks.org](http://en.wikibooks.org)
[2552 http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ALazy-lump](http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ALazy-lump)
[2553 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreggreg](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreggreg)
[2554 http://en.wikibooks.org](http://en.wikibooks.org)
[2555 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento)
[2556 http://en.wikibooks.org](http://en.wikibooks.org)
[2557 http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASpiderworm](http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASpiderworm)
[2558 http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen](http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAverieZen)
[2559 http://en.wikibooks.org](http://en.wikibooks.org)

1172	b:en:User:Spiderworm ²⁵⁶⁰	PD
1173	Original uploader was Allefant ²⁵⁶¹ at en.wikibooks ²⁵⁶²	
1174	Romain Behar. Original uploader was Romainbehar ²⁵⁶³ at en.wikibooks ²⁵⁶⁴	PD
1175	John Whelan. Original uploader was Johnwhelan ²⁵⁶⁵ at en.wikibooks ²⁵⁶⁶ . Later version(s) were uploaded by Adrignola ²⁵⁶⁷ at en.wikibooks ²⁵⁶⁸ .	
1176		
1177	me	GFDL
1178	Toni Grappa ²⁵⁶⁹	GFDL
1179	Blender Foundation	GPL
1180	Guillaume22 ²⁵⁷⁰	GFDL
1181	Original uploader was Gabio ²⁵⁷¹ at en.wikibooks ²⁵⁷²	PD
1182	Boyage ²⁵⁷³ . Original uploader was Boyage ²⁵⁷⁴ at en.wikibooks ²⁵⁷⁵	
1183	Blender Foundation	GPL
1184	Romain Behar. Original uploader was Romainbehar ²⁵⁷⁶ at en.wikibooks ²⁵⁷⁷	PD
1185	Original uploader was Robpers2003 ²⁵⁷⁸ at en.wikibooks ²⁵⁷⁹	
1186	User:Toni Grappa ²⁵⁸⁰	
1187	Original uploader was Totophe64 ²⁵⁸¹ at en.wikibooks ²⁵⁸²	
1188	Blender Foundation	GPL
1189	Boyage ²⁵⁸³ . Original uploader was Boyage ²⁵⁸⁴ at en.wikibooks ²⁵⁸⁵	
1190	SoylentGreen ²⁵⁸⁶	cc-by-2.5

2560 <http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASpiderworm>
2561 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAllefant>
2562 <http://en.wikibooks.org>
2563 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>
2564 <http://en.wikibooks.org>
2565 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AJohnwhelan>
2566 <http://en.wikibooks.org>
2567 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
2568 <http://en.wikibooks.org>
2569 http://en.wikibooks.org/wiki/User%3AToni_grappa
2570 <http://en.wikibooks.org/wiki/User%3AGuillaume22>
2571 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>
2572 <http://en.wikibooks.org>
2573 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
2574 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
2575 <http://en.wikibooks.org>
2576 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>
2577 <http://en.wikibooks.org>
2578 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARobpers2003>
2579 <http://en.wikibooks.org>
2580 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2581 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>
2582 <http://en.wikibooks.org>
2583 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
2584 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
2585 <http://en.wikibooks.org>
2586 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

1191	Original uploader was Totophe64 ²⁵⁸⁷ at en.wikibooks ²⁵⁸⁸	
1192	Blender Foundation	GPL
1193	Original uploader was SchmS ²⁵⁸⁹ at en.wikibooks ²⁵⁹⁰	GFDL
1194	Blender Foundation	GPL
1195	SoylentGreen ²⁵⁹¹	GFDL
1196	User:Toni Grappa ²⁵⁹²	
1197	User:Toni Grappa ²⁵⁹³	
1198	me	GFDL
1199	Blender Foundation	GPL
1200	User:Toni Grappa ²⁵⁹⁴	
1201	User:Toni Grappa ²⁵⁹⁵	
1202	Toni Grappa; Veröffentlichung mit freundlicher Genehmigung von Paddy - blend.polis ²⁵⁹⁶	
1203	User:Bastique ²⁵⁹⁷ , User:Ramac ²⁵⁹⁸ et al.	
1204	SaphireS	cc-by-sa-3.0
1205	Blender Foundation	GPL
1206	Original uploader was Mrfelis ²⁵⁹⁹ at en.wikibooks ²⁶⁰⁰	
1207	Original uploader was MikaYuodas ²⁶⁰¹ at fr.wikiversity ²⁶⁰²	GFDL
1208	me	GFDL
1209	Toni Grappa ²⁶⁰³	
1210	Original uploader was Stalepanda ²⁶⁰⁴ at en.wikibooks ²⁶⁰⁵	GFDL
1211	Toni Grappa ²⁶⁰⁶	GFDL
1212	SoylentGreen ²⁶⁰⁷	GFDL
1213	Original uploader was Sumoncil ²⁶⁰⁸ at en.wikibooks ²⁶⁰⁹ . Later version(s) were uploaded by Adrignola ²⁶¹⁰ at en.wikibooks ²⁶¹¹ .	

2587 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

2588 <http://en.wikibooks.org>

2589 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASchmS>

2590 <http://en.wikibooks.org>

2591 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2592 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2593 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2594 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2595 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2596 http://en.wikibooks.org/wiki/User%3AToni_grappa

2597 <http://en.wikibooks.org/wiki/User%3ABastique>

2598 <http://en.wikibooks.org/wiki/User%3ARamac>

2599 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>

2600 <http://en.wikibooks.org>

2601 <http://en.wikibooks.org/wiki/%3Awikiversity%3Afr%3AUser%3AMikaYuodas>

2602 <http://fr.wikiversity.org>

2603 http://en.wikibooks.org/wiki/User%3AToni_grappa

2604 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AStalepanda>

2605 <http://en.wikibooks.org>

2606 http://en.wikibooks.org/wiki/User%3AToni_grappa

2607 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2608 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>

2609 <http://en.wikibooks.org>

2610 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2611 <http://en.wikibooks.org>

1214	Original uploader was Darkonc ²⁶¹² at en.wikibooks ²⁶¹³ . Later version(s) were uploaded by Argento ²⁶¹⁴ at en.wikibooks ²⁶¹⁵ .	GFDL
1215	Frogulis ²⁶¹⁶	PD
1216	Guillaume22 ²⁶¹⁷	GFDL
1217	Blender Foundation	GPL
1218	Original uploader was Greensweater ²⁶¹⁸ at en.wikibooks ²⁶¹⁹	
1219	Guillaume22 ²⁶²⁰	GFDL
1220	Original uploader was Stalepanda ²⁶²¹ at en.wikibooks ²⁶²²	GFDL
1221	SoylentGreen ²⁶²³	GFDL
1222	User:Toni Grappa ²⁶²⁴	
1223	SaphireS	cc-by-sa-3.0
1224	User:Toni Grappa ²⁶²⁵	
1225	SoylentGreen ²⁶²⁶	GFDL
1226	User:Toni Grappa ²⁶²⁷	
1227	blender Foundation	PD
1228		
1229	User:Toni Grappa ²⁶²⁸	
1230	User:Toni Grappa ²⁶²⁹	
1231	SoylentGreen ²⁶³⁰	GFDL
1232	User:Toni Grappa ²⁶³¹	
1233	pearts ²⁶³²	
1234	Blender Foundation	GPL
1235	Original uploader was Gabio ²⁶³³ at en.wikibooks ²⁶³⁴	
1236	SVG version was created by User:Grunt ²⁶³⁵ and cleaned up by 3247 ²⁶³⁶ , based on the earlier PNG version, created by Reidab ²⁶³⁷ .	

2612 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>
2613 <http://en.wikibooks.org>
2614 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>
2615 <http://en.wikibooks.org>
2616 <http://en.wikibooks.org/wiki/User%3AFrogulis>
2617 <http://en.wikibooks.org/wiki/User%3AGuillaume22>
2618 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>
2619 <http://en.wikibooks.org>
2620 <http://en.wikibooks.org/wiki/User%3AGuillaume22>
2621 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AStalepanda>
2622 <http://en.wikibooks.org>
2623 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2624 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2625 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2626 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2627 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2628 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2629 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2630 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2631 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2632 <http://en.wikibooks.org/wiki/User%3Apearts>
2633 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>
2634 <http://en.wikibooks.org>
2635 <http://en.wikibooks.org/wiki/User%3AGrunt>
2636 <http://en.wikibooks.org/wiki/User%3A3247>
2637 <http://en.wikibooks.org/wiki/%3Am%3AUser%3AReidab>

1237	Mangarm ²⁶³⁸	GPL
1238	SoylentGreen ²⁶³⁹	GFDL
1239	Original uploader was Megaloman ²⁶⁴⁰ at en.wikibooks ²⁶⁴¹	
1240	SoylentGreen ²⁶⁴²	GPL
1241	Blender Foundation	GPL
1242	Original uploader was Darkonc ²⁶⁴³ at en.wikibooks ²⁶⁴⁴ . Later version(s) were uploaded by Argento ²⁶⁴⁵ at en.wikibooks ²⁶⁴⁶ .	GFDL
1243	SoylentGreen ²⁶⁴⁷	GPL
1244		GFDL
1245	SoylentGreen ²⁶⁴⁸	GFDL
1246	SoylentGreen ²⁶⁴⁹	PD
1247	Toni Grappa ²⁶⁵⁰	GFDL
1248		GFDL
1249	SoylentGreen ²⁶⁵¹	GPL
1250	Toni Grappa ²⁶⁵²	GFDL
1251		
1252	SoylentGreen ²⁶⁵³	GFDL
1253	Toni Grappa ²⁶⁵⁴	GFDL
1254	SaphireS	cc-by-sa-3.0
1255	User:Toni Grappa ²⁶⁵⁵	
1256	User:Toni Grappa ²⁶⁵⁶	
1257	unknown	GFDL
1258	Romain Behar. Original uploader was Romainbehar ²⁶⁵⁷ at en.wikibooks ²⁶⁵⁸	PD
1259	User:Toni Grappa ²⁶⁵⁹	
1260	SoylentGreen ²⁶⁶⁰	GFDL

²⁶³⁸ <http://en.wikibooks.org/wiki/User%3AManagarm>

²⁶³⁹ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁴⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>

²⁶⁴¹ <http://en.wikibooks.org>

²⁶⁴² <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁴³ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

²⁶⁴⁴ <http://en.wikibooks.org>

²⁶⁴⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁶⁴⁶ <http://en.wikibooks.org>

²⁶⁴⁷ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁴⁸ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁴⁹ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁵⁰ http://en.wikibooks.org/wiki/User%3AToni_grappa

²⁶⁵¹ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁵² http://en.wikibooks.org/wiki/User%3AToni_grappa

²⁶⁵³ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

²⁶⁵⁴ http://en.wikibooks.org/wiki/User%3AToni_grappa

²⁶⁵⁵ <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

²⁶⁵⁶ <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

²⁶⁵⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

²⁶⁵⁸ <http://en.wikibooks.org>

²⁶⁵⁹ <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

²⁶⁶⁰ <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

1261	Boyage ²⁶⁶¹ . Original uploader was Boyage ²⁶⁶² at en.wikibooks ²⁶⁶³	
1262		GFDL
1263	SoylentGreen ²⁶⁶⁴	GFDL
1264		
1265	SoylentGreen ²⁶⁶⁵	
1266	SoylentGreen ²⁶⁶⁶	GFDL
1267	Original uploader was Spiderworm ²⁶⁶⁷ at en.wikibooks ²⁶⁶⁸	GPL
1268	Original uploader was Gabio ²⁶⁶⁹ at en.wikibooks ²⁶⁷⁰	PD
1269	SoylentGreen ²⁶⁷¹	GFDL
1270	User:Toni Grappa ²⁶⁷²	
1271	Soylentgreen ²⁶⁷³	GFDL
1272	Blender Foundation	GPL
1273	SoylentGreen ²⁶⁷⁴	GPL
1274	Binky1206 ²⁶⁷⁵	PD
1275	SoylentGreen ²⁶⁷⁶	GFDL
1276	Original uploader was Bgks ²⁶⁷⁷ at en.wikibooks ²⁶⁷⁸	
1277	Mangarm ²⁶⁷⁹	GPL
1278	Original uploader was Mrfelis ²⁶⁸⁰ at en.wikibooks ²⁶⁸¹	
1279	User:Toni Grappa ²⁶⁸²	
1280	Original uploader was Megaloman ²⁶⁸³ at en.wikibooks ²⁶⁸⁴	GPL
1281	Original uploader was Sumoncil ²⁶⁸⁵ at en.wikibooks ²⁶⁸⁶ . Later version(s) were uploaded by Adrignola ²⁶⁸⁷ at en.wikibooks ²⁶⁸⁸ .	

2661 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
2662 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABoyage>
2663 <http://en.wikibooks.org>
2664 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2665 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2666 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2667 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASpiderworm>
2668 <http://en.wikibooks.org>
2669 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>
2670 <http://en.wikibooks.org>
2671 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2672 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2673 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>
2674 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2675 <http://en.wikibooks.org/wiki/User%3ABinky1206>
2676 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>
2677 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABgks>
2678 <http://en.wikibooks.org>
2679 <http://en.wikibooks.org/wiki/User%3AMangarm>
2680 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMrfelis>
2681 <http://en.wikibooks.org>
2682 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>
2683 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AMegaloman>
2684 <http://en.wikibooks.org>
2685 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>
2686 <http://en.wikibooks.org>
2687 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>
2688 <http://en.wikibooks.org>

1282	Original uploader was Sumoncil ²⁶⁸⁹ at en.wikibooks ²⁶⁹⁰ . Later version(s) were uploaded by Adrignola ²⁶⁹¹ at en.wikibooks ²⁶⁹² .	
1283	me	GFDL
1284	Original uploader was Gabio ²⁶⁹³ at en.wikibooks ²⁶⁹⁴	PD
1285	Original uploader was Greensweater ²⁶⁹⁵ at en.wikibooks ²⁶⁹⁶	
1286	Toni Grappa ²⁶⁹⁷	GFDL
1287	Blender Foundation	GPL
1288	Original uploader was MikaYuodas ²⁶⁹⁸ at fr.wikiversity ²⁶⁹⁹	GFDL
1289	Yhevhe	GFDL
1290	Toni Grappa ²⁷⁰⁰	GFDL
1291	SoylentGreen ²⁷⁰¹	GPL
1292	Thomas Dinges	GFDL
1293	WebBird ²⁷⁰²	GFDL
1294	b:en>User:Spiderworm ²⁷⁰³	PD
1295	Original uploader was Argento ²⁷⁰⁴ at en.wikibooks ²⁷⁰⁵	
1296	Romain Behar. Original uploader was Romainbehar ²⁷⁰⁶ at en.wikibooks ²⁷⁰⁷	PD
1297	Guillaume22 ²⁷⁰⁸	GFDL
1298	Original uploader was Bgks ²⁷⁰⁹ at en.wikibooks ²⁷¹⁰	PD
1299	Original uploader was Badalia ²⁷¹¹ at en.wikibooks ²⁷¹²	cc-by-sa-2.5
1300	Original uploader was Darkonc ²⁷¹³ at en.wikibooks ²⁷¹⁴ . Later version(s) were uploaded by Argento ²⁷¹⁵ at en.wikibooks ²⁷¹⁶ .	GFDL
1301	Original uploader was Totophe64 ²⁷¹⁷ at en.wikibooks ²⁷¹⁸	

2689 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>

2690 <http://en.wikibooks.org>

2691 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2692 <http://en.wikibooks.org>

2693 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGabio>

2694 <http://en.wikibooks.org>

2695 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreensweater>

2696 <http://en.wikibooks.org>

2697 http://en.wikibooks.org/wiki/User%3AToni_grappa

2698 <http://en.wikibooks.org/wiki/%3Awikiversity%3Afr%3AUser%3AMikaYuodas>

2699 <http://fr.wikiversity.org>

2700 http://en.wikibooks.org/wiki/User%3AToni_grappa

2701 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2702 <http://en.wikibooks.org/wiki/User%3AWebBird>

2703 <http://en.wikibooks.org/wiki/%3Ab%3Aen%3AUser%3ASpiderworm>

2704 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2705 <http://en.wikibooks.org>

2706 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ARomainbehar>

2707 <http://en.wikibooks.org>

2708 <http://en.wikibooks.org/wiki/User%3AGuillaume22>

2709 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABgks>

2710 <http://en.wikibooks.org>

2711 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABadalia>

2712 <http://en.wikibooks.org>

2713 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

2714 <http://en.wikibooks.org>

2715 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2716 <http://en.wikibooks.org>

2717 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATotophe64>

2718 <http://en.wikibooks.org>

1302	Blender Foundation	GPL
1303	User:Toni Grappa ²⁷¹⁹	
1304	Soylentgreen ²⁷²⁰	GFDL
1305	Soylentgreen ²⁷²¹	GFDL
1306	Original uploader was Sumoncil ²⁷²² at en.wikibooks ²⁷²³ . Later version(s) were uploaded by Adrignola ²⁷²⁴ at en.wikibooks ²⁷²⁵ .	
1307	SoylentGreen ²⁷²⁶	GFDL
1308	Guillaume22 ²⁷²⁷	GFDL
1309	SoylentGreen ²⁷²⁸	PD
1310	Guillaume22 ²⁷²⁹	GFDL
1311	Original uploader was Sumoncil ²⁷³⁰ at en.wikibooks ²⁷³¹ . Later version(s) were uploaded by Adrignola ²⁷³² at en.wikibooks ²⁷³³ .	
1312	Original uploader was ZeroOne ²⁷³⁴ at en.wikibooks ²⁷³⁵	
1313	Original uploader was Sumoncil ²⁷³⁶ at en.wikibooks ²⁷³⁷ . Later version(s) were uploaded by Adrignola ²⁷³⁸ at en.wikibooks ²⁷³⁹ .	
1314	Original uploader was ZeroOne ²⁷⁴⁰ at en.wikibooks ²⁷⁴¹	
1315	Original uploader was Darkonc ²⁷⁴² at en.wikibooks ²⁷⁴³ . Later version(s) were uploaded by Argento ²⁷⁴⁴ at en.wikibooks ²⁷⁴⁵ .	GFDL
1316		PD
1317		PD
1318	Original uploader was Gregggreg ²⁷⁴⁶ at en.wikibooks ²⁷⁴⁷	
1319	Original uploader was SchmS ²⁷⁴⁸ at en.wikibooks ²⁷⁴⁹	GFDL

2719 <http://en.wikibooks.org/wiki/User%3AToni%20Grappa>

2720 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>

2721 <http://en.wikibooks.org/wiki/User%3ASoylentgreen>

2722 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>

2723 <http://en.wikibooks.org>

2724 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2725 <http://en.wikibooks.org>

2726 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2727 <http://en.wikibooks.org/wiki/User%3AGuillaume22>

2728 <http://en.wikibooks.org/wiki/User%3ASoylentGreen>

2729 <http://en.wikibooks.org/wiki/User%3AGuillaume22>

2730 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>

2731 <http://en.wikibooks.org>

2732 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2733 <http://en.wikibooks.org>

2734 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne>

2735 <http://en.wikibooks.org>

2736 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASumoncil>

2737 <http://en.wikibooks.org>

2738 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2739 <http://en.wikibooks.org>

2740 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AZeroOne>

2741 <http://en.wikibooks.org>

2742 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

2743 <http://en.wikibooks.org>

2744 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2745 <http://en.wikibooks.org>

2746 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AGreggreg>

2747 <http://en.wikibooks.org>

2748 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASchmS>

2749 <http://en.wikibooks.org>

1320	Original uploader was SchmS ²⁷⁵⁰ at en.wikibooks ²⁷⁵¹	GFDL
1321	Original uploader was SatanClaus ²⁷⁵² at en.wikibooks ²⁷⁵³	
1322	Original uploader was SatanClaus ²⁷⁵⁴ at en.wikibooks ²⁷⁵⁵	
1323	Original uploader was Argento ²⁷⁵⁶ at en.wikibooks ²⁷⁵⁷	
1324	Original uploader was Argento ²⁷⁵⁸ at en.wikibooks ²⁷⁵⁹	
1325	Original uploader was SatanClaus ²⁷⁶⁰ at en.wikibooks ²⁷⁶¹	
1326	Original uploader was N comme Nul ²⁷⁶² at en.wikibooks ²⁷⁶³	
1327	Original uploader was N comme Nul ²⁷⁶⁴ at en.wikibooks ²⁷⁶⁵	
1328	Original uploader was SchmS ²⁷⁶⁶ at en.wikibooks ²⁷⁶⁷	GFDL
1329	Original uploader was Darkonc ²⁷⁶⁸ at en.wikibooks ²⁷⁶⁹ . Later version(s) were uploaded by Argento ²⁷⁷⁰ at en.wikibooks ²⁷⁷¹ .	GFDL
1330	Original uploader was SatanClaus ²⁷⁷² at en.wikibooks ²⁷⁷³ . Later versions were uploaded by Argento ²⁷⁷⁴ at en.wikibooks ²⁷⁷⁵ .	GPL
1331	Original uploader was Darkonc ²⁷⁷⁶ at en.wikibooks ²⁷⁷⁷ . Later version(s) were uploaded by Argento ²⁷⁷⁸ at en.wikibooks ²⁷⁷⁹ .	GFDL
1332	Original uploader was Darkonc ²⁷⁸⁰ at en.wikibooks ²⁷⁸¹ . Later version(s) were uploaded by Argento ²⁷⁸² at en.wikibooks ²⁷⁸³ .	GFDL

²⁷⁵⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASchmS>

²⁷⁵¹ <http://en.wikibooks.org>

²⁷⁵² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASatanClaus>

²⁷⁵³ <http://en.wikibooks.org>

²⁷⁵⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASatanClaus>

²⁷⁵⁵ <http://en.wikibooks.org>

²⁷⁵⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁷⁵⁷ <http://en.wikibooks.org>

²⁷⁵⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁷⁵⁹ <http://en.wikibooks.org>

²⁷⁶⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASatanClaus>

²⁷⁶¹ <http://en.wikibooks.org>

²⁷⁶² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AN%20comme%20Nul>

²⁷⁶³ <http://en.wikibooks.org>

²⁷⁶⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AN%20comme%20Nul>

²⁷⁶⁵ <http://en.wikibooks.org>

²⁷⁶⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASchmS>

²⁷⁶⁷ <http://en.wikibooks.org>

²⁷⁶⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

²⁷⁶⁹ <http://en.wikibooks.org>

²⁷⁷⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁷⁷¹ <http://en.wikibooks.org>

²⁷⁷² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASatanClaus>

²⁷⁷³ <http://en.wikibooks.org>

²⁷⁷⁴ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁷⁷⁵ <http://en.wikibooks.org>

²⁷⁷⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

²⁷⁷⁷ <http://en.wikibooks.org>

²⁷⁷⁸ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁷⁷⁹ <http://en.wikibooks.org>

²⁷⁸⁰ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

²⁷⁸¹ <http://en.wikibooks.org>

²⁷⁸² <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

²⁷⁸³ <http://en.wikibooks.org>

1333	Original uploader was SatanClaus ²⁷⁸⁴ at en.wikibooks ²⁷⁸⁵ . Later versions were uploaded by Argento ²⁷⁸⁶ at en.wikibooks ²⁷⁸⁷ .	
1334	Original uploader was Darkonc ²⁷⁸⁸ at en.wikibooks ²⁷⁸⁹ . Later version(s) were uploaded by Argento ²⁷⁹⁰ at en.wikibooks ²⁷⁹¹ .	
1335	Original uploader was Darkonc ²⁷⁹² at en.wikibooks ²⁷⁹³ . Later version(s) were uploaded by Argento ²⁷⁹⁴ at en.wikibooks ²⁷⁹⁵ .	GFDL
1336	Original uploader was Darkonc ²⁷⁹⁶ at en.wikibooks ²⁷⁹⁷ . Later version(s) were uploaded by Argento ²⁷⁹⁸ at en.wikibooks ²⁷⁹⁹ .	GFDL
1337	Original uploader was Darkonc ²⁸⁰⁰ at en.wikibooks ²⁸⁰¹ . Later version(s) were uploaded by Argento ²⁸⁰² at en.wikibooks ²⁸⁰³ .	GFDL
1338	Original uploader was Argento ²⁸⁰⁴ at en.wikibooks ²⁸⁰⁵ .	PD
1339	DDD3x ²⁸⁰⁶	GPL
1340	DDD3x ²⁸⁰⁷	GPL
1341	DDD3x ²⁸⁰⁸	GPL
1342	DDD3x ²⁸⁰⁹	GPL
1343	DDD3x ²⁸¹⁰	GPL
1344	DDD3x ²⁸¹¹	GPL
1345	DDD3x ²⁸¹²	GPL
1346	DDD3x ²⁸¹³	GPL
1347	DDD3x ²⁸¹⁴	PD
1348		cc-by-sa-2.0

2784 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASatanClaus>

2785 <http://en.wikibooks.org>

2786 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2787 <http://en.wikibooks.org>

2788 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

2789 <http://en.wikibooks.org>

2790 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2791 <http://en.wikibooks.org>

2792 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

2793 <http://en.wikibooks.org>

2794 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2795 <http://en.wikibooks.org>

2796 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

2797 <http://en.wikibooks.org>

2798 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2799 <http://en.wikibooks.org>

2800 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ADarkonc>

2801 <http://en.wikibooks.org>

2802 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2803 <http://en.wikibooks.org>

2804 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AArgento>

2805 <http://en.wikibooks.org>

2806 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2807 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2808 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2809 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2810 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2811 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2812 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2813 <http://en.wikibooks.org/wiki/User%3ADDD3x>

2814 <http://en.wikibooks.org/wiki/User%3ADDD3x>

1349	Original uploader was Bootaleg ²⁸¹⁵ at en.wikibooks ²⁸¹⁶	cc-by-sa-2.0
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
0	Loremus Ipsemus	None
1375	Blender Foundation	GPL
1376	Blender Foundation	GPL
1377	Original uploader was Nuclearpidgeon ²⁸¹⁷ at en.wikibooks ²⁸¹⁸	
1378	Blender Foundation	GPL
1379	Michał Sapalski (username: sapalskimichal)	PD
1380	Michał Sapalski (username: sapalskimichal)	PD
1381	Michał Sapalski (username: sapalskimichal)	PD
1382	Michał Sapalski (username: sapalskimichal)	PD
1383	Michał Sapalski (username: sapalskimichal)	PD
1384	Michał Sapalski (username: sapalskimichal)	PD
1385	Michał Sapalski (username: sapalskimichal)	PD
1386	Michał Sapalski (username: sapalskimichal)	PD
1387	Rosver ²⁸¹⁹	cc-by-sa-3.0
1388	Rosver ²⁸²⁰	PD
1389	blender Foundation	PD
1390	Blender Foundation	PD
1391	Blender Foundation	GPL
1392	Blender Foundation	GPL

²⁸¹⁵ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ABootaleg>

²⁸¹⁶ <http://en.wikibooks.org>

²⁸¹⁷ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ANuclearpidgeon>

²⁸¹⁸ <http://en.wikibooks.org>

²⁸¹⁹ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸²⁰ <http://en.wikibooks.org/wiki/User%3ARosver>

1393	Blender Foundation	GPL
1394	Blander Foundation	GPL
1395	Blender Foundation	GPL
1396	Blender Foundation	GPL
1397	Blender 3d Screen Shots	GPL
1398	Blender Foundation	GPL
1399	Blender Foundation	GPL
1400	Blender Foundation	GPL
1401	Blender Foundation	GPL
1402	Blender Foundation	GPL
1403	Blender Foundation	GPL
1404	Blender Foundation	GPL
1405	<Blender Foundation	GPL
1406	Blender Foundation	GPL
1407	Blender Foundation	GPL
1408	Blender Foundation	GPL
1409	Blender Foundation	GPL
1410	Blender Foundation	GPL
1411	Blender Foundation	GPL
1412	Blender Foundation	GPL
1413	Rosver ²⁸²¹	PD
1414		PD
1415	Original uploader was Quantum Anomaly ²⁸²² at en.wikibooks ²⁸²³	GFDL
1416	Original uploader was Quantum Anomaly ²⁸²⁴ at en.wikibooks ²⁸²⁵	GFDL
1417	Original uploader was Quantum Anomaly ²⁸²⁶ at en.wikibooks ²⁸²⁷	GFDL
1418	Original uploader was Quantum Anomaly ²⁸²⁸ at en.wikibooks ²⁸²⁹	GFDL
1419	Original uploader was Quantum Anomaly ²⁸³⁰ at en.wikibooks ²⁸³¹	GFDL
1420	Original uploader was Quantum Anomaly ²⁸³² at en.wikibooks ²⁸³³	GFDL
1421	Original uploader was Quantum Anomaly ²⁸³⁴ at en.wikibooks ²⁸³⁵	GFDL
1422	Original uploader was Quantum Anomaly ²⁸³⁶ at en.wikibooks ²⁸³⁷	GFDL
1423	Original uploader was Quantum Anomaly ²⁸³⁸ at en.wikibooks ²⁸³⁹	GFDL

2821 <http://en.wikibooks.org/wiki/User%3ARosver>

2822 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2823 <http://en.wikibooks.org>

2824 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2825 <http://en.wikibooks.org>

2826 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2827 <http://en.wikibooks.org>

2828 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2829 <http://en.wikibooks.org>

2830 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2831 <http://en.wikibooks.org>

2832 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2833 <http://en.wikibooks.org>

2834 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2835 <http://en.wikibooks.org>

2836 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2837 <http://en.wikibooks.org>

2838 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2839 <http://en.wikibooks.org>

1424	Original uploader was Quantum Anomaly ²⁸⁴⁰ at en.wikibooks ²⁸⁴¹	GFDL
1425	Original uploader was Quantum Anomaly ²⁸⁴² at en.wikibooks ²⁸⁴³	GFDL
1426	Original uploader was Quantum Anomaly ²⁸⁴⁴ at en.wikibooks ²⁸⁴⁵	GFDL
1427	Original uploader was Quantum Anomaly ²⁸⁴⁶ at en.wikibooks ²⁸⁴⁷	GFDL
1428	Original uploader was Quantum Anomaly ²⁸⁴⁸ at en.wikibooks ²⁸⁴⁹	GFDL
1429	Original uploader was Quantum Anomaly ²⁸⁵⁰ at en.wikibooks ²⁸⁵¹ . Later version(s) were uploaded by Adrignola ²⁸⁵² at en.wikibooks ²⁸⁵³ .	GFDL
1430	Original uploader was Soeb ²⁸⁵⁴ at en.wikibooks ²⁸⁵⁵	
1431	Fabrice TIERCELIN ²⁸⁵⁶	GFDL
1432	Original uploader was Soeb ²⁸⁵⁷ at en.wikibooks ²⁸⁵⁸	
1433	Original uploader was Soeb ²⁸⁵⁹ at en.wikibooks ²⁸⁶⁰	
1434	Fabrice TIERCELIN ²⁸⁶¹	GFDL
1435		PD
1436		PD
1437		PD
1438		PD
1439		PD
1440		PD
1441		PD
1442		PD
1443	othman ahmad	GPL
1444	Othman Ahmad	GPL
1445	Original uploader was Tjb0607 ²⁸⁶² at en.wikibooks ²⁸⁶³	
1446	Original uploader was Tjb0607 ²⁸⁶⁴ at en.wikibooks ²⁸⁶⁵	

2840 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2841 <http://en.wikibooks.org>

2842 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2843 <http://en.wikibooks.org>

2844 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2845 <http://en.wikibooks.org>

2846 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2847 <http://en.wikibooks.org>

2848 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2849 <http://en.wikibooks.org>

2850 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AQuantum%20Anomaly>

2851 <http://en.wikibooks.org>

2852 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3AAdrignola>

2853 <http://en.wikibooks.org>

2854 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

2855 <http://en.wikibooks.org>

2856 <http://en.wikibooks.org/wiki/User%3Aftiercel>

2857 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

2858 <http://en.wikibooks.org>

2859 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ASoeb>

2860 <http://en.wikibooks.org>

2861 <http://en.wikibooks.org/wiki/User%3Aftiercel>

2862 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATjb0607>

2863 <http://en.wikibooks.org>

2864 <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATjb0607>

2865 <http://en.wikibooks.org>

1447	Original uploader was Tjb0607 ²⁸⁶⁶ at en.wikibooks ²⁸⁶⁷	
1448	tjb0607	
1449	User:Tjb0607 ²⁸⁶⁸	
1450	Tjb0607 ²⁸⁶⁹	cc-by-sa-3.0
1451	Ldo ²⁸⁷⁰	
1452	Blender Foundation	GPL
1453	Blender Foundation	GPL
1454	Blender Foundation	GPL
1455	Rosver ²⁸⁷¹	PD
1456	Rosver ²⁸⁷²	PD
1457	Rosver ²⁸⁷³	PD
1458	Rosver ²⁸⁷⁴	PD
1459	Rosver ²⁸⁷⁵	PD
1460	Rosver ²⁸⁷⁶	PD
1461	Rosver ²⁸⁷⁷	PD
1462	Rosver ²⁸⁷⁸	PD
1463	Rosver ²⁸⁷⁹	
1464	Rosver ²⁸⁸⁰	PD
1465	Rosver ²⁸⁸¹	PD
1466	Rosver ²⁸⁸²	PD
1467	Rosver ²⁸⁸³	PD
1468	Rosver ²⁸⁸⁴	PD
1469	Rosver ²⁸⁸⁵	PD
1470	Rosver ²⁸⁸⁶	PD
1471	Rosver ²⁸⁸⁷	PD
1472	Rosver ²⁸⁸⁸	PD
1473	Rosver ²⁸⁸⁹	PD

²⁸⁶⁶ <http://en.wikibooks.org/wiki/%3Awikibooks%3Aen%3AUser%3ATjb0607>

²⁸⁶⁷ <http://en.wikibooks.org>

²⁸⁶⁸ <http://en.wikibooks.org/wiki/User%3ATjb0607>

²⁸⁶⁹ <http://en.wikibooks.org/wiki/User%3ATjb0607>

²⁸⁷⁰ <http://en.wikibooks.org/wiki/User%3ALdo>

²⁸⁷¹ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷² <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷³ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷⁴ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷⁵ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷⁶ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷⁷ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷⁸ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁷⁹ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁰ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸¹ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸² <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸³ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁴ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁵ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁶ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁷ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁸ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁸⁸⁹ <http://en.wikibooks.org/wiki/User%3ARosver>

1474	Rosver ²⁸⁹⁰	PD
1475	Rosver ²⁸⁹¹	PD
1476	Rosver ²⁸⁹²	PD
1477	Rosver ²⁸⁹³	PD
1478	Rosver ²⁸⁹⁴	PD
1479	Rosver ²⁸⁹⁵	PD
1480	Rosver ²⁸⁹⁶	PD
1481	Rosver ²⁸⁹⁷	PD
1482	Rosver ²⁸⁹⁸	PD
1483	Rosver ²⁸⁹⁹	PD
1484	Rosver ²⁹⁰⁰	PD
1485	Rosver ²⁹⁰¹	PD
1486	Rosver ²⁹⁰²	PD
1487	Rosver ²⁹⁰³	PD
1488	Rosver ²⁹⁰⁴	PD
1489	Rosver ²⁹⁰⁵	PD
1490	Rosver ²⁹⁰⁶	PD
1491	Rosver ²⁹⁰⁷	PD
1492	Rosver ²⁹⁰⁸	PD
1493	Rosver ²⁹⁰⁹	PD
1494	Rosver ²⁹¹⁰	PD
1495	Rosver ²⁹¹¹	PD
1496	Rosver ²⁹¹²	PD
1497	Rosver ²⁹¹³	PD
1498	Rosver ²⁹¹⁴	PD
1499	Rosver ²⁹¹⁵	PD

2890 <http://en.wikibooks.org/wiki/User%3ARosver>
 2891 <http://en.wikibooks.org/wiki/User%3ARosver>
 2892 <http://en.wikibooks.org/wiki/User%3ARosver>
 2893 <http://en.wikibooks.org/wiki/User%3ARosver>
 2894 <http://en.wikibooks.org/wiki/User%3ARosver>
 2895 <http://en.wikibooks.org/wiki/User%3ARosver>
 2896 <http://en.wikibooks.org/wiki/User%3ARosver>
 2897 <http://en.wikibooks.org/wiki/User%3ARosver>
 2898 <http://en.wikibooks.org/wiki/User%3ARosver>
 2899 <http://en.wikibooks.org/wiki/User%3ARosver>
 2900 <http://en.wikibooks.org/wiki/User%3ARosver>
 2901 <http://en.wikibooks.org/wiki/User%3ARosver>
 2902 <http://en.wikibooks.org/wiki/User%3ARosver>
 2903 <http://en.wikibooks.org/wiki/User%3ARosver>
 2904 <http://en.wikibooks.org/wiki/User%3ARosver>
 2905 <http://en.wikibooks.org/wiki/User%3ARosver>
 2906 <http://en.wikibooks.org/wiki/User%3ARosver>
 2907 <http://en.wikibooks.org/wiki/User%3ARosver>
 2908 <http://en.wikibooks.org/wiki/User%3ARosver>
 2909 <http://en.wikibooks.org/wiki/User%3ARosver>
 2910 <http://en.wikibooks.org/wiki/User%3ARosver>
 2911 <http://en.wikibooks.org/wiki/User%3ARosver>
 2912 <http://en.wikibooks.org/wiki/User%3ARosver>
 2913 <http://en.wikibooks.org/wiki/User%3ARosver>
 2914 <http://en.wikibooks.org/wiki/User%3ARosver>
 2915 <http://en.wikibooks.org/wiki/User%3ARosver>

1500	Rosver ²⁹¹⁶	PD
1501	Rosver ²⁹¹⁷	PD
1502	Rosver ²⁹¹⁸	PD
1503	Rosver ²⁹¹⁹	PD
1504	Rosver ²⁹²⁰	PD
1505	Rosver ²⁹²¹	PD
1506	Rosver ²⁹²²	PD
1507	Rosver ²⁹²³	PD
1508	Rosver ²⁹²⁴	PD
1509	Rosver ²⁹²⁵	PD
1510	Rosver ²⁹²⁶	PD
1511	Rosver ²⁹²⁷	PD
1512	Rosver ²⁹²⁸	PD
1513	Rosver ²⁹²⁹	PD
1514	Rosver ²⁹³⁰	PD
1515	Rosver ²⁹³¹	PD
1516	Rosver ²⁹³²	PD
1517	Rosver ²⁹³³	PD
1518	Rosver ²⁹³⁴	PD
1519	Rosver ²⁹³⁵	PD
1520	Rosver ²⁹³⁶	PD
1521	Rosver ²⁹³⁷	PD
1522	Rosver ²⁹³⁸	PD
1523	Rosver ²⁹³⁹	PD
1524	Rosver ²⁹⁴⁰	PD
1525	Rosver ²⁹⁴¹	PD

2916 <http://en.wikibooks.org/wiki/User%3ARosver>
2917 <http://en.wikibooks.org/wiki/User%3ARosver>
2918 <http://en.wikibooks.org/wiki/User%3ARosver>
2919 <http://en.wikibooks.org/wiki/User%3ARosver>
2920 <http://en.wikibooks.org/wiki/User%3ARosver>
2921 <http://en.wikibooks.org/wiki/User%3ARosver>
2922 <http://en.wikibooks.org/wiki/User%3ARosver>
2923 <http://en.wikibooks.org/wiki/User%3ARosver>
2924 <http://en.wikibooks.org/wiki/User%3ARosver>
2925 <http://en.wikibooks.org/wiki/User%3ARosver>
2926 <http://en.wikibooks.org/wiki/User%3ARosver>
2927 <http://en.wikibooks.org/wiki/User%3ARosver>
2928 <http://en.wikibooks.org/wiki/User%3ARosver>
2929 <http://en.wikibooks.org/wiki/User%3ARosver>
2930 <http://en.wikibooks.org/wiki/User%3ARosver>
2931 <http://en.wikibooks.org/wiki/User%3ARosver>
2932 <http://en.wikibooks.org/wiki/User%3ARosver>
2933 <http://en.wikibooks.org/wiki/User%3ARosver>
2934 <http://en.wikibooks.org/wiki/User%3ARosver>
2935 <http://en.wikibooks.org/wiki/User%3ARosver>
2936 <http://en.wikibooks.org/wiki/User%3ARosver>
2937 <http://en.wikibooks.org/wiki/User%3ARosver>
2938 <http://en.wikibooks.org/wiki/User%3ARosver>
2939 <http://en.wikibooks.org/wiki/User%3ARosver>
2940 <http://en.wikibooks.org/wiki/User%3ARosver>
2941 <http://en.wikibooks.org/wiki/User%3ARosver>

1526	Rosver ²⁹⁴²	PD
1527	Rosver ²⁹⁴³	PD
1528	Rosver ²⁹⁴⁴	PD
1529	Rosver ²⁹⁴⁵	PD
1530	Rosver ²⁹⁴⁶	PD
1531	Rosver ²⁹⁴⁷	PD
1532	Rosver ²⁹⁴⁸	PD
1533	Rosver ²⁹⁴⁹	PD
1534	Rosver ²⁹⁵⁰	
1535	Rosver ²⁹⁵¹	
1536	Rosver ²⁹⁵²	
1537	Rosver ²⁹⁵³	
1538		GFDL
1539	PHGCOM • derivative work: Rosver ²⁹⁵⁴ (talk ²⁹⁵⁵)	GFDL
1540	Blender Foundation	GPL
1541	blender foundation	GPL
1542	Blender Foundation	GPL
1543	Blender Foundation	GPL
1544	Rosver ²⁹⁵⁶	PD
1545	Kome111 ²⁹⁵⁷	GFDL
1546		GPL
1547		GPL
1548		GPL
1549		GPL
1550		GPL
1551		GPL
1552		GPL

²⁹⁴² <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴³ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴⁴ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴⁵ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴⁶ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴⁷ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴⁸ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁴⁹ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵⁰ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵¹ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵² <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵³ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵⁴ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵⁵ <http://en.wikibooks.org/wiki/User%20talk%3ARosver>

²⁹⁵⁶ <http://en.wikibooks.org/wiki/User%3ARosver>

²⁹⁵⁷ <http://en.wikibooks.org/wiki/User%3AKome111>

244. Licenses

244.1. GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <<http://fsf.org/>>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program—to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assure copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow. TERMS AND CONDITIONS 0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays "Appropriate Legal Notices" to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion. 1. Source Code.

The "source code" for a work means the preferred form of the work for making modifications to it. "Object code" means any non-source form of a work.

A "Standard Interface" means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The "System Libraries" of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A "Major Component", in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The "Corresponding Source" for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those

activities. However, it does not include the work's System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work. 2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable and exclusive provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary. 3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures. 4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee. 5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

- The work must carry prominent notices stating that you modified it, and giving a relevant date. * b) The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices". * c) You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it. * d) If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate. 6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

- Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange. * b) Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge. * c) Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b. * d) Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements. * e) Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work that the User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying. 7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

- Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or * b) Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or * c) Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of that material be marked in reasonable ways as different from the original version; or * d) Limiting the use for publicity purposes of names of licensors or authors of the material; or * e) Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or * f) Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way. 8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10. 9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so. 10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An "entity transaction" is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party's predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it. 11. Patents.

A "contributor" is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor's "contributor version".

A contributor's "essential patent claims" are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, "control" includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor's essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a "patent license" is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To "grant" such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. "Knowingly relying" means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient's use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law. 12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program. 13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such. 14. Revised Versions of This License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright

holder as a result of your choosing to follow a later version. 15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION. 16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR

LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. 17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty, and each file should have at least the "copyright" line and a pointer to where the full notice is found.

A section "Entitled XYZ" means a named subunit of the Document whose title either is precisely XYZ or contains XYZ in parentheses following text that translates XYZ in another language. (Here XYZ stands for a specific section name mentioned below, such as "Acknowledgements", "Dedications", "Endorsements", or "History"). To "Preserve the Title" of such a section when you modify the Document means that it remains a section "Entitled XYZ" according to this definition.

The Document may include Warranty Disclaimers next to the notice which states that this License applies to the Document. These Warranty Disclaimers are considered to be included by reference in this License, but only as regards disclaiming warranties; any other implication that these Warranty Disclaimers may have is void and has no effect on the meaning of this License. 2. VERBATIM COPYING

You may copy and distribute the Document in any medium, either commercially or noncommercially, provided that this License, the copyright notices, and the license notice saying this License applies to the Document are reproduced in all copies, and that you add no other conditions whatsoever to those of this License. You may not use technical measures to obstruct or control the reading or further copying of the copies you make or distribute. However, you may accept compensation in exchange for copies. If you distribute a large enough number of copies you must also follow the conditions in section 3.

You may also lend copies, under the same conditions stated above, and you may publicly display copies. 3. COPYING IN QUANTITY

If you publish printed copies (or copies in media that commonly have printed covers) of the Document, numbering more than 100, and the Document's license notice requires Cover Texts, you must enclose the copies in covers that carry, clearly and legibly, all these Cover Texts: Front-Cover Texts on the front cover, and Back-Cover Texts on the back cover. Both covers must also clearly and legibly identify you as the publisher of these copies. The front cover must present the full title with all words of the title equally prominent and visible. You may add other material on the covers in addition. Copying with changes limited to the covers, as long as they preserve the title of the Document and satisfy these conditions, can be treated as verbatim copying in other respects.

If the required texts for either cover are too voluminous to fit legibly, you should put the first ones listed (as many as fit reasonably) on the actual cover, and continue the rest on adjacent pages.

If you publish or distribute Opaque copies of the Document numbering more than 100, you must either include a machine-readable Transparent copy along with each Opaque copy, or state in or with each Opaque copy a computer network location from which the general network-using public has access to download using public-standard network protocols a complete Transparent copy of the Document, free of added material. If you use the latter option, you must take reasonably prudent steps, when you begin distribution of Opaque copies in quantity, to ensure that this Transparent copy will remain thus accessible at the stated location until at least one year after the last time you distribute an Opaque copy (directly or through your agents or retailers) of that edition to the public.

If it is requested, but not required, that you contact the authors of the Document will before redistributing any large number of copies, to give them a chance to provide you with an updated version of the Document. 4. MODIFICATIONS

You may copy and distribute a Modified Version of the Document under the conditions of sections 2 and 3 above, provided that you release the Modified Version under precisely this License, with the Modified Version filling the role of the Document, thus licensing distribution and modification of the Modified Version to whoever possesses a copy of it. In addition, you must do these things in the Modified Version:

* A. Use in the Title Page (and on the covers, if any) a title distinct from that of the Document, and from those of previous versions (which should, if there were any, be listed in the History section of the Document). You may use the same title as a previous version if the original publisher of that version gives permission. * B. List on the Title Page, as authors, one or more persons or entities responsible for authorship of the modifications in the Modified Version, together with at least five of the principal authors of the Document (all of its principal authors, if it has fewer than five), unless they release you from this requirement. * C. State on the Title page the name of the publisher of the Modified Version, as the publisher. * D. Preserve all the copyright notices of the Document. * E. Add an appropriate copyright notice for your modifications adjacent to the other copyright notices. * F. Include, immediately after the copyright notices, a license notice giving the public permission to use the Modified Version under the terms of this License, in the form shown in the Addendum below. * G. Preserve in that license notice the full lists of Invariant Sections and required Cover Texts given in the Document's license notice. * H. Include an unaltered copy of this License. * I. Preserve the section Entitled "History", Preserve its Title, and add to it an item stating at least the title, year, new authors, and publisher of the Modified Version as given on the Title Page. If there is no section Entitled "History" in the Document, create one stating the title, year, authors, and publisher of the Document as given on its Title Page, then add an item describing the Modified Version as stated in the previous section. * J. Preserve the network location, if any, given in the Document for public access to a Transparent copy of the Document, and likewise the network locations given in the Document for previous versions if they were used. These may be placed in the "History" section. You may omit a network location for a work that was published at least four years before the Document itself, or if the

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see <http://www.gnu.org/licenses/>.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

original publisher of the version it refers to gives permission. * K. For any section Entitled "Acknowledgements" or "Dedications", Preserve the Title of the section, and preserve in the section all the substance and tone of each of the contributor acknowledgments and/or dedications given therein. * L. Preserve all the Invariant Sections of the Document, unaltered in their text and in their titles. Section numbers or the equivalent are not considered part of the section titles. * M. Delete any section Entitled "Endorsements". Such a section may not be included in the Modified Version. * N. Do not retittle any existing section to be Entitled "Endorsements" or to conflict in title with any Invariant Section. * O. Preserve any Warranty Disclaimers.

If the Modified Version includes new front-matter sections or appendices that qualify as Secondary Sections and contain no material copied from the Document, you may at your option designate some or all of these sections as invariant. To do this, add their titles to the list of Invariant Sections in the Modified Version's license notice. These titles must be distinct from any other section titles.

You may add a section Entitled "Endorsements", provided it contains nothing but endorsements of your Modified Version by various parties—for example, statements of peer review or of that the text has been approved by an organization as the authoritative definition of a standard.

You may add a passage of up to five words as a Front-Cover Text, and a passage of up to 25 words as a Back-Cover Text, to the end of the list of Cover Texts in the Modified Version. Only one passage of Front-Cover Text and one of Back-Cover Text may be added by (or through arrangements made by) any one entity. If the Document already includes a cover text for the same cover, previously added by you or by arrangement made by the same entity you are acting on behalf of, you may not add another; but you may replace the old one, on explicit permission from the previous publisher that added the old one.

The author(s) and publisher(s) of the Document do not by this license give permission to use their names for publicity for or to assert or imply endorsement of any Modified Version. 5. COMBINING DOCUMENTS

You may combine the Document with other documents released under this License, under the terms defined in section 4 above for modified versions, provided that you include in the combination all of the Invariant Sections of all of the original documents, unmodified, and list them all as Invariant Sections of your combined work in its license notice, and that you preserve all their Warranty Disclaimers.

The combined work need only contain one copy of this License, and multiple identical Invariant Sections may be replaced with a single copy. If there are multiple Invariant Sections with the same name but different contents, make the title of each such section unique by adding at the end of it, in parentheses, the name of the original author or publisher of that section if known, or else a unique number. Make the same adjustment to the section titles in the list of Invariant Sections in the license notice of the combined work.

In the combination, you must combine any sections Entitled "History" in the various original documents, forming one section Entitled "History"; likewise combine any sections Entitled "Acknowledgements", and any sections Entitled "Dedications". You must delete all sections Entitled "Endorsements". 6. COLLECTIONS OF DOCUMENTS

You may make a collection consisting of the Document and other documents released under this License, and replace the individual copies of this License in the various documents with a single copy that is included in the collection, provided that you follow the rules of this License for verbatim copying of each of the documents in all other respects.

You may extract a single document from such a collection, and distribute it individually under this License, provided you insert a copy of this License into the extracted document, and follow this License in all other respects regarding verbatim copying of that document. 7. AGGREGATION WITH INDEPENDENT WORKS

A compilation of the Document or its derivatives with other separate and independent documents or works, in or on a volume of a storage or distribution medium, is called an "aggregate" if the copyright resulting from the compilation is not used to limit the legal rights of the compilation's users beyond what the individual works permit. When the Document is included in an aggregate, this License does not apply to the other works in the aggregate which are not themselves derivative works of the Document.

If the Cover Text requirement of section 3 is applicable to these copies of the Document, then if the Document is less than one half of the entire aggregate, the Document's Cover Texts may be placed on covers that bracket the Document within the aggregate, or the electronic equivalent of covers if the Document is in electronic form. Otherwise they must appear on printed covers that bracket the whole aggregate. 8. TRANSLATION

Translation is considered a kind of modification, so you may distribute translations of the Document under the terms of section 4. Replacing Invariant Sections with translations requires special permission from their copyright holders, but you may include translations of some or all Invariant Sections in addition to the original versions of these Invariant Sections. You may include a translation of this License, and all the license notices in the Document, and any Warranty Disclaimers, provided that you also include the original English version of this License and the original versions of those notices and disclaimers. In case of a disagreement between the translation and the original version of this License or a notice or disclaimer, the original version will prevail.

"The Library" refers to a covered work governed by this License, other than an Application or a Combined Work as defined below.

An "Application" is any work that makes use of an interface provided by the Library, but which is not otherwise based on the Library. Defining a subclass of a class defined by the Library is deemed a mode of using an interface provided by the Library.

<program> Copyright (C) <year> <name of author>. This program comes with ABSOLUTELY NO WARRANTY; for details type 'show w'. This is free software, and you are welcome to redistribute it under certain conditions; type 'show c' for details.

The hypothetical commands 'show w' and 'show c' should show the appropriate parts of the General Public License. Of course, your program's commands might be different; for a GUI interface, you would use an "about box".

You should also get your employer (if you work as a programmer) or school, if any, to sign a "copyright disclaimer" for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see <http://www.gnu.org/licenses/>.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read <http://www.gnu.org/philosophy/why-not-lgpl.html>.

If a section in the Document is Entitled "Acknowledgements", "Dedications", or "History", the requirement (section 4) to Preserve its Title (section 1) will typically require changing the actual title. 9. TERMINATION

You may not copy, modify, sublicense, or distribute the Document except as expressly provided under this License. Any attempt otherwise to copy, modify, sublicense, or distribute it is void, and will automatically terminate your rights under this License.

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, receipt of a copy of some or all of the same material does not give you any rights to use it. 10. FUTURE REVISIONS OF THIS LICENSE

The Free Software Foundation may publish new, revised versions of the GNU Free Documentation License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns. See <http://www.gnu.org/copyleft/>.

Each version of the License is given a distinguishing version number. If the Document specifies that a particular numbered version of this License "or any later version" applies to it, you have the option of following the terms and conditions either of that specified version or of any later version that has been published (not as a draft) by the Free Software Foundation. If the Document does not specify a version number of this License, you may choose any version ever published (not as a draft) by the Free Software Foundation. If the Document specifies that a proxy can decide which future versions of this License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Document. 11. RELICENSING

"Massive Multiauthor Collaboration Site" (or "MMC Site") means any World Wide Web server that publishes copyrightable works and also provides prominent facilities for anybody to edit those works. A public wiki that anybody can edit is an example of such a server. A "Massive Multi-author Collaboration" (or "MMC") contained in the site means any set of copyrightable works thus published on the MMC site.

"CC-BY-SA" means the Creative Commons Attribution-Share Alike 3.0 license published by Creative Commons Corporation, a not-for-profit corporation with a principal place of business in San Francisco, California, as well as future copyleft versions of that license published by that same organization.

"Incorporate" means to publish or republish a Document, in whole or in part, as part of another Document.

An MMC is "eligible for relicensing" if it is licensed under this License, and if all works that were first published under this License somewhere other than this MMC, and subsequently incorporated in whole or in part into the MMC, (1) had no cover texts or invariant sections, and (2) were thus incorporated prior to November 1, 2008.

The operator of an MMC Site may republish an MMC contained in the site under CC-BY-SA on the same site at any time before August 1, 2009, provided the MMC is eligible for relicensing. ADDENDUM: How to use this License for your documents

To use this License in a document you have written, include a copy of the License in the document and put the following copyright and license notices just after the title page:

Copyright (C) YEAR YOUR NAME. Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free Software Foundation, with no Invariant Sections, no Front-Cover Texts, and no Back-Cover Texts. A copy of the license is included in the section entitled "GNU Free Documentation License".

If you have Invariant Sections, Front-Cover Texts and Back-Cover Texts, replace the "with ... Texts." line with this:

with the Invariant Sections being LIST THEIR TITLES, with the Front-Cover Texts being LIST, and with the Back-Cover Texts being LIST.

If you have Invariant Sections without Cover Texts, or some other combination of the three, merge those two alternatives to suit the situation.

If your document contains nontrivial examples of program code, we recommend releasing these examples in parallel under your choice of free software license, such as the GNU General Public License, to permit their use in free software.

244.2. GNU Free Documentation License

Version 1.3, 3 November 2008

Copyright © 2000, 2001, 2002, 2007, 2008 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. 0. PREAMBLE

The purpose of this License is to make a manual, textbook, or other functional and useful document "free" in the sense of freedom: to assure everyone the effective freedom to copy and redistribute it, with or without modifying it, either commercially or noncommercially. Secondarily, this License preserves for the author and publisher a way to get their work, while not being considered responsible for modifications made by others.

This License is a kind of "copyleft", which means that derivative works of the document must themselves be free in the same sense. It complements the GNU General Public License, which is a copyleft license designed for free software.

We have designed this License in order to use it for manuals for free software, because free software needs free documentation: a free program should come with manuals providing the same freedoms that the software does. But this License is not limited to software manuals; it can be used for any textual work, regardless of subject matter or whether it is published as a printed book. We recommend this License principally for works whose purpose is instruction or reference. 1. APPLICABILITY AND DEFINITIONS

This License applies to any manual or other work, in any medium, that contains a notice placed by the copyright holder saying it can be distributed under the terms of this License. Such a notice grants a world-wide, royalty-free license, unlimited in duration, to use that work under the conditions stated herein. The "Document", below, refers to any such manual or work. Any member of the public is a licensee, and is addressed as "you". You accept the license if you copy, modify or distribute the work in a way requiring permission under copyright law.

A "Modified Version" of the Document means any work containing the Document or a portion of it, either copied verbatim, or with modifications and/or translated into another language.

A "Secondary Section" is a named appendix or a front-matter section of the Document that deals exclusively with the relationship of the publishers or authors of the Document to the Document's overall subject (or to related matters) and contains nothing that could fall directly within that overall subject. (Thus, if the Document is in part a textbook of mathematics, a Secondary Section may not explain any mathematics.) The relationship could be a matter of historical connection with the subject or with related matters, or of legal, commercial, philosophical, ethical or political position regarding them.

The "Invariant Sections" are certain Secondary Sections whose titles are designated, as being those of Invariant Sections, in the notice that says that the Document is released under this License. If a section does not fit the above definition of Secondary then it is not allowed to be designated as Invariant. The Document may contain zero Invariant Sections. If the Document does not identify any Invariant Sections then there are none.

The "Cover Texts" are certain short passages of text that are listed, as Front-Cover Texts or Back-Cover Texts, in the notice that says that the Document is released under this License. A Front-Cover Text may be at most 5 words, and a Back-Cover Text may be at most 25 words.

A "Transparent" copy of the Document means a machine-readable copy, represented in a format whose specification is available to the general public, that is suitable for revising the document straightforwardly with generic text editors or (for images composed of pixels) generic paint programs or (for drawings) some widely available drawing editor, and that is suitable for input to text formatters or for automatic translation to a variety of formats suitable for input to text formatters. A copy made in an otherwise Transparent file format whose markup, or absence of markup, has been arranged to thwart or discourage subsequent modification by readers is not Transparent. An image format is not Transparent if used for any substantial amount of text. A copy that is not "Transparent" is called "Opaque".

Examples of suitable formats for Transparent copies include plain ASCII without markup, Texinfo input format, LaTeX input format, SGML or XML using a publicly available DTD, and standard-conforming simple HTML, PostScript or PDF designed for human modification. Examples of transparent image formats include PNG, XCF and JPG. Opaque formats include proprietary formats that can be read and edited only by proprietary word processors, SGML or XML for which the DTD and/or processing tools are not generally available, and the machine-generated HTML, PostScript or PDF produced by some word processors for output purposes only.

The "Title Page" means, for a printed book, the title page itself, plus following pages as are needed to hold, legibly, the material this License requires to appear in the title page. For works in formats which do not have any title page as such, "Title Page" means the text near the most prominent appearance of the work's title, preceding the beginning of the body of the text.

The "publisher" means any person or entity that distributes copies of the Document to the public.

244.3. GNU Lesser General Public License

GNU LESSER GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. <http://fsf.org/>

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

This version of the GNU Lesser General Public License incorporates the terms and conditions of version 3 of the GNU General Public License, supplemented by the additional permissions listed below. 0. Additional Definitions.

As used herein, "this License" refers to version 3 of the GNU Lesser General Public License, and the "GNU GPL" refers to version 3 of the GNU General Public License.

The "Corresponding Application Code" for a Combined Work means the object code and/or source code for the Application, including any data and utility programs needed for reproducing the Combined Work from the Application, but excluding the System Libraries of the Combined Work. 1. Exception to Section 3 of the GNU GPL.

You may convey a covered work under sections 3 and 4 of this License without being bound by section 3 of the GNU GPL. 2. Conveying Modified Versions.

If you modify a copy of the Library, and, in your modifications, a facility refers to a function or data to be supplied by an Application that uses the facility (other than as an argument passed when the facility is invoked), then you may convey a copy of the modified version:

* a) under this License, provided that you make a good faith effort to ensure that, in the event an Application does not supply the function or data, the facility still operates, and performs whatever part of its purpose remains meaningful, or * b) under the GNU GPL, with none of the additional permissions of this License applicable to that copy.

3. Object Code Incorporating Material from Library Header Files.

The object code form of an Application may incorporate material from a header file that is part of the Library. You may convey such object code

under terms of your choice, provided that, if the incorporated material is not limited to numerical parameters, data structure layouts and accessors, or small macros, inline functions and templates (ten or fewer lines in length), you do both of the following:

* a) Give prominent notice with each copy of the object code that the Library is used in it and that the Library and its use are covered by this License. * b) Accompany the object code with a copy of the GNU GPL and this license document.

4. Combined Works.

You may convey a Combined Work under terms of your choice that, taken together, effectively do not restrict modification of the portions of the Library contained in the Combined Work and reverse engineering for debugging such modifications, if you also do each of the following:

* a) Give prominent notice with each copy of the Combined Work that the Library is used in it and that the Library and its use are covered by this License. * b) Accompany the Combined Work with a copy of the GNU GPL and this license document. * c) For a Combined Work that displays copyright notices during execution, include the copyright notice for the Library among these notices, as well as a reference directing the user to the copies of the GNU GPL and this license document. * d) Do one of the following: o 0) Convey the Minimal Corresponding Source under the terms of this License, and the Corresponding Application Code in a form suitable for, and under terms

that permit, the user to recombine or relink the Application with a modified version of the Linked Version to produce a modified Combined Work, in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source. o 1) Use a suitable shared library mechanism for linking with the Library. A suitable mechanism is one that (a) uses at run time a copy of the Library already present on the user's computer system, and (b) will operate properly with a modified version of the Library that is interface-compatible with the Linked Version. * e) Provide Installation Information, but only if you would otherwise be required to provide such information under section 6 of the GNU GPL, and only to the extent that such information is necessary to install and execute a modified version of the Combined Work produced by recombining or relinking the Application with a modified version of the Linked Version. (If you use option 4d0, the Installation Information must accompany the Minimal Corresponding Source and Corresponding Application Code. If you use option 4d1, you must provide the Installation Information in the manner specified by section 6 of the GNU GPL for conveying Corresponding Source.)

5. Combined Libraries.

You may place library facilities that are a work based on the Library side by side in a single library together with other library facilities that are not Applications and are not covered by this License, and convey such a combined library under terms of your choice, if you do both of the following:

* a) Accompany the combined library with a copy of the same work based on the Library, uncombined with any other library facilities, conveyed under the terms of this License. * b) Give prominent notice with the combined library that part of it is a work based on the Library, and explaining where to find the accompanying uncombined form of the same work.

6. Revised Versions of the GNU Lesser General Public License.

The Free Software Foundation may publish revised and/or new versions of the GNU Lesser General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Library as you received it specifies that a certain numbered version of the GNU Lesser General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that published version or of any later version published by the Free Software Foundation. If the Library as you received it does not specify a version number of the GNU Lesser General Public License, you may choose any version of the GNU Lesser General Public License ever published by the Free Software Foundation.

If the Library as you received it specifies that a proxy can decide whether future versions of the GNU Lesser General Public License shall apply, that proxy's public statement of acceptance of any version is permanent authorization for you to choose that version for the Library.