

SLUD 2006

**El sistema de computación simbólica, Máxima**  
In memoriam William Schelter

Robert Dodier  
Proyecto Máxima

Libero este documento por el  
GNU General Public License version 2

## ¿Por qué la computación simbólica?

Recurremos a computación por lo general para resolver algún problema de “mundo real”

Una vez que se resuelve, normalmente queremos resolver el mismo problema otra vez con datos poco diferentes

La computación simbólica es un atentado de resolver toda una clase de problemas en una vez

Por eso dejamos que algunos variables en el problema no tengan valores fijos

Entonces tenemos que computar sin que sepamos valores de todos variables

## **Máxima, un sistema de computación simbólica**

Máxima es un programa interactivo para la computación simbólico

Modo de computación es simbólica normalmente, hace computaciones numéricas si lo pide

Puede manejar aritmética, polinomios, solución de ecuaciones lineales y algunas polinomias, cálculo diferencial y integral, ecuaciones diferenciales, conjuntos, vectores y matrices, números primos, números complejos, y gráficos

Máxima es un proyecto de software libre con el hospedaje de SourceForge

Ya tiene un equipo internacional de desarrollo

## Unos ejemplos

Miremos a unos ejemplos sencillos que indican algo de Máxima

Una operación de polinomio:

```
(%i1) a : 2;
```

```
(%o1)          2
```

```
(%i2) expand ((a + b)^10);
```

## Unos ejemplos ...

```
(%i1) a : 2;
```

```
(%o1)          2
```

```
(%i2) expand ((a + b)^10);
```

```
          10          9          8          7          6
(%o2) b  + 20 b  + 180 b  + 960 b  + 3360 b
        5          4          3          2
+ 8064 b  + 13440 b  + 15360 b  + 11520 b
+ 5120 b + 1024
```

## Unos ejemplos ...

Una operación de matriz:

```
(%i1) M : matrix ([%pi, %e], [17, 29]);
```

```
      [ %pi  %e ]
```

```
(%o1)      [      ]
```

```
      [ 17   29 ]
```

```
(%i2) invert (M);
```

## Unos ejemplos ...

```
(%i1) M : matrix ([%pi, %e], [17, 29]);  
          [ %pi  %e ]
```

```
(%o1)      [          ]  
          [ 17    29 ]
```

```
(%i2) invert (M);
```

```
          [          29          %e          ]  
          [ ----- - ----- ]  
          [ 29 %pi - 17 %e    29 %pi - 17 %e ]  
(%o2) [          ]  
          [          17          %pi          ]  
          [ - ----- ----- ]  
          [ 29 %pi - 17 %e    29 %pi - 17 %e ]
```

## Unos ejemplos ...

Una integración:

```
(%i1) integrate (exp (- x * u), u, 0, inf);
```



## Unos ejemplos ...

```
(%i1) integrate (exp (- x * u), u, 0, inf);
```

Is x positive, negative, or zero?

```
pos;
```

```
1
```

```
(%o1)
```

```
-
```

```
x
```

## Historia breve de Máxima

1968: Inició MACSYMA como parte del Proyecto MAC (“Cognición Ayudada por Máquina”). Fundado por el Departamento de Energía (DOE) del EE.UU.

MACSYMA = “Matemáticas simbólicas del Proyecto MAC”

1968–1982: Desarrollo de MACSYMA en el Instituto de Tecnología de Massachusetts (MIT)

En esos años, MACSYMA era proyecto de investigación académica, para el beneficio de los estudiantes y profesores

1982: Entrega de MACSYMA de MIT a DOE. Esa versión era llamada “DOE-MACSYMA”

1982–1996 Desarrollo comercial de MACSYMA

1982–2001: Desarrollo de DOE-MACSYMA por William Schelter (Universidad de Texas)

1998: Schelter obtuvo permisión de DOE de liberar DOE-MACSYMA con licencia GPL

2001-presente: Desarrollo de DOE-MACSYMA (ya llamada Máxima) como proyecto de hospedaje SourceForge

## Características generales de Máxima

Toda cosa es una expresión. Es una manera de decir “código = datos” en Máxima.

Máxima = colección de funciones para transformar expresiones

La actitud “laissez faire”. Si Máxima no sabe que hacer con una expresión, no hace nada (por la mayor parte). Por eso es muy fácil extender las habilidades de Máxima.

Distinción entre evaluación y simplificación

Evaluación = sustitución de valores por símbolos y invocación de funciones, Simplificación = sustitución de expresiones por equivalentes

## Toda cosa es una expresión

Expresiones algebraicas como  $x + y$ ,  $\sin^2 x$ ,  $\sum_{i=1}^n f(i)$

También sumaciones y integrales

También secuencias de expresiones (**block**)

También objetos compuestos como  $[a, b, c]$  (lista),  $1, 2, 3$  (conjunto), **matrix**( $[a, b], [1, 2]$ ) (matriz)

También funciones  $f(x) := x^2$

También el bucle **for** y el condicional **if – then – else**

La implicación es que trabajar con Máxima es manipular expresiones

## **La actitud “laissez faire”**

Máxima intenta tratar cada expresión con todas leyes matemáticas que sabe

¿Qué sucede cuando Máxima no sabe más?

Si Máxima no sabe que hacer con una expresión, lo deja quedar como está (en la gran mayoría de casos)

Así deja la posibilidad de cambiar algo y tratarla otra vez

El usuario o programador puede inventar nuevas funciones que empiezan donde Máxima se deja

## Máxima y Lisp

Máxima es escrito en Common Lisp

Es muy fácil caer desde Máxima hasta Lisp

Se puede entrar en el ambiente Lisp y ver todos objetos Lisp

Funciones y objetos definidos en Lisp se puede ver en Máxima,  
y viceversa

## **Lisp como lenguaje de implementación**

De principio, Máxima fue escrito en MacLisp, un dialecto bastante anticuado ahora

Ya el código fuente es Common Lisp, pero no explota las habilidades de CL (desarrollado en los años 80 y 90)

...énfasis en la lista de propiedades (SYMBOL-PLIST)

...no mucho uso de paquetes

...muchos variables globales

...no uso de números complejos o racionales de Lisp

...la cadena en Máxima es un símbolo en Lisp

Es difícil hallar todos los códigos que manejar a un operador; la manera de programación no necesita la localización de códigos



## Lenguaje de programación de Máxima

Herencia de Lisp, con forma de Algol

**map, apply, lambda** – Funciones Lisp que aparecen en Máxima

Ámbito dinámico (originalmente el sólo tipo de ámbito)

Máxima es un “Lisp-2”: con el mismo símbolo, se puede definir múltiples objetos (variable, función, otros)

No tiene sistema de paquetes

No tiene estructuras ni objetos (en el sentido de C++ o Java)

El lenguaje de Máxima es sencillo — es fácil expresar ideas sencillas

## Fortalezas de Máxima

Máxima hace fácil resolver problemas sencillos, y hace posible resolver problemas complejos

Máxima es muy general, y aumentamos su generalidad continuamente

Máxima trata a un gran número de sujetos básicos

...y tiene también un gran colección de paquetes adicionales

Hemos integrados varios otros programas de Lisp y de Fortran

Hay mucho interés en Máxima hoy día, y mucha actividad de desarrollo

## Faltas de Máxima

Asunción que hay alguna persona en el teclado – gran problema para integrar con otro sistema

No se puede retraer todos reglas

No se puede borrar todo valores sin reiniciar

Sistema de declaraciones es débil

Objetos compuestos (matrices, conjuntos, listas) tienen que ser concretos, no simbólicos

Falta de uniformidad — Máxima es el producto de muchas manos por muchos años

Falta un sistema de paquetes

Ámbito dinámico (ámbito lexical sería mejor)

## Comparación con otros sistemas

### *Otros sistemas generales*

Mathematica, Maple, MuPAD — más o menos la misma extensión de matemáticas como Máxima; más funciones, menos errores, mejores interfaces

Yacas (SL) — más o menos la misma extensión; menos funciones (proyecto mucho más joven que Máxima)

Axiom (SL) – sistema de categorías estrictas (“strongly typed”). Certamente no “laissez faire”

### *Otros sistemas para sujetos específicos*

PARI/GP (SL) (teoría de números); GAP (SL), Magma (teoría de grupos); ACL2 (SL) (pruebas automáticas)

## *Otros sistemas numéricos*

Octave (SL), Matlab, Scilab (numéricos generales); R (SL), S-Plus (estadística); Mathcad (énfasis en el interfaz cuaderno)

Hay mucho interés ahora de integrar sistemas para construir un sistema que tenga más habilidad que algún solo sistema e.g. SAGE integra a GAP, SINGULAR, PARI/GP, etc; Matlab integra a Maple

Problema más grande para integración de sistemas es falta de uniformidad de comunicaciones

Se ha propuesto el uso de XML en alguna forma (e.g. MathML); pero aún se acuerda en el formato, también necesita acordar en la interpretación de un mensaje

## **Máxima como fenómeno social**

Tenemos ahora un equipo diverso y internacional

Como otros proyectos de software libre, lo que pasa es más o menos lo que quiere los desarrolladores

Era unos años atrás un plan de desarrollo, pero no lo hemos seguido cuidadosamente

En particular, ha sido mucho interés de interfaces gráficas, igual que revisión y traducción del manual de referencia

También hemos trabajado en corregir errores y aumentar funciones existentes

Máxima es una casa de muchos cámaras — se puede trabajar en lo que se acuerda con sí mismo

## **A donde vamos con Máxima**

No tenemos un plan fijo. Algunas cosas que podríamos hacer:

Aumentar la documentación (otras traducciones y nuevos textos)

Interfaz “cuaderno” para el usuario

Interfaz programático más sencillo

Corregir errores en funciones de límites y integrales

Corregir otros errores

Hacer más fácil la división de funciones (así que se puede omitir o reemplazar alguna parte)

Máxima como sistema de propósito general – el lenguaje “C” para las matemáticas simbólicas y numéricas

## Por qué viní a Máxima

Era trabajando en problemas de estadística y decisión (análisis de sobrevivencia)

Encontré Máxima buscando una solución. Todavía quiero resolver problemas de tipo similar

Lo que hace más fácil esta tarea: (1) algoritmos más fuertes para integrar; (2) interfaz cuaderno; (3) integración con códigos numéricos o generación de tal códigos

Quisiera resolver problemas más o menos prácticos, y parece que Máxima es un buen vehículo de eso

Quisiera ver paquetes de extensión para diversos sujetos matemáticos (problemas de ingeniería, estadística, comercio, las ciencias, etc etc)



## **William Schelter**

Mil gracias otra vez a William Schelter

Lo que vio él fue que Máxima merece la pena de desarrollo y difusión

Conseguió la licencia que garantiza que Máxima queda libre y vivo aún en su ausencia

Otorgó sus esfuerzos a la publica y a nosotros para que las disfrutemos

Con esta inspiración, dedicamos de nuevo nuestras propias esfuerzos para el beneficio de todos

## Recursos generales

Portada del proyecto (versión castellano): <http://maxima.sourceforge.net/es>

Página del proyecto en SourceForge: <http://sourceforge.net/projects/maxima>

Documentación (versión castellano):

<http://maxima.sourceforge.net/es/docs.shtml>

Errores: [http://sourceforge.net/tracker/?group\\_id=4933&atid=104933](http://sourceforge.net/tracker/?group_id=4933&atid=104933)

Correos electronicos (versión castellano):

<http://lists.sourceforge.net/lists/listinfo/maxima-lang-es>

Obtener el software:

[http://sourceforge.net/project/showfiles.php?group\\_id=4933](http://sourceforge.net/project/showfiles.php?group_id=4933)

Versiones para diversos sistemas:

<http://maxima.sourceforge.net/wiki/index.php/Maxima%20ports>

Máxima wiki: <http://maxima.sourceforge.net/wiki>