

Ingres 10.0

Installation Guide

INGRES

ING-10-IN-01

This Documentation is for the end user's informational purposes only and may be subject to change or withdrawal by Ingres Corporation ("Ingres") at any time. This Documentation is the proprietary information of Ingres and is protected by the copyright laws of the United States and international treaties. It is not distributed under a GPL license. You may make printed or electronic copies of this Documentation provided that such copies are for your own internal use and all Ingres copyright notices and legends are affixed to each reproduced copy.

You may publish or distribute this document, in whole or in part, so long as the document remains unchanged and is disseminated with the applicable Ingres software. Any such publication or distribution must be in the same manner and medium as that used by Ingres, e.g., electronic download via website with the software or on a CD-ROM. Any other use, such as any dissemination of printed copies or use of this documentation, in whole or in part, in another publication, requires the prior written consent from an authorized representative of Ingres.

To the extent permitted by applicable law, INGRES PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IN NO EVENT WILL INGRES BE LIABLE TO THE END USER OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USER OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF INGRES IS EXPRESSLY ADVISED OF SUCH LOSS OR DAMAGE.

The manufacturer of this Documentation is Ingres Corporation.

For government users, the Documentation is delivered with "Restricted Rights" as set forth in 48 C.F.R. Section 12.212, 48 C.F.R. Sections 52.227-19(c)(1) and (2) or DFARS Section 252.227-7013 or applicable successor provisions.

Copyright © 2010 Ingres Corporation. All Rights Reserved.

Ingres, OpenROAD, and EDBC are registered trademarks of Ingres Corporation. All other trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

Contents

Chapter 1: Understanding Installation Considerations **9**

An Ingres Instance.....	9
Instance ID	9
System Administrator Account	10
Ingres Files	11
Location of Ingres Files	12
File Location Guidelines.....	13
II_SYSTEM (System Files) Location Guidelines	13
II_DATABASE (Database Files) Location Guidelines.....	14
II_CHECKPOINT, II_JOURNAL, II_DUMP (Checkpoint, Journal, Dump Files) Location Guidelines	15
II_WORK (Temporary Work Files) Location Guidelines.....	16
Transaction (and Backup) Log Location Guidelines.....	16
Use of Ingres Locations.....	17
Location Use During Normal Operations	17
Location Use During OLTP Operations	18
Location Use During DSS or OLAP Operations	19
Location Use During Checkpointing.....	20
Location Use During Recovery from a Checkpoint	21
Location Use During Replay of Journals.....	22
Sample Disk Configurations	22
Four-Disk DBMS Server Configuration.....	23
Three-Disk DBMS Server Configuration	23
Two-Disk DBMS Server Configuration	23
One-Disk DBMS Server Configuration	23
Client-only Disk Configuration	24
NFS Client Disk Configuration (UNIX)	24
World Region and Time Zone	24
Time Zone Names.....	24
Character Set	26
Date Alias Setting	28
Installation Parameters	28
General Installation Parameters	28
DBMS Server Installation Parameters	29
Ingres Servers.....	31
Valid Computer, Directory, and User Names.....	32

Chapter 2: Installing Ingres **33**

How You Install Ingres.....	33
-----------------------------	----

Installing Ingres for Linux	33
Install Ingres for Linux Using the Installation Wizard.....	34
Installing Ingres for Linux at the Command Line	34
Install Ingres for Windows.....	41
Separately Installed Components	41
Advanced Component Selection	41
Perform a Silent Install on Windows	42
Install Ingres for UNIX.....	44
Install Ingres for OpenVMS.....	44
How You Access the Instance on Linux and UNIX.....	46
Configuration Type Parameter Settings.....	47

Chapter 3: Starting Ingres **49**

Start the Ingres Installation on Linux and UNIX.....	49
Startup and Shutdown on Windows	50
Start and Stop Ingres with Ingres Visual Manager	51
Start and Stop Ingres with the Ingres Service Manager	52
Start the Ingres Installation on OpenVMS	53
Getting Started with Databases and Applications	53

Chapter 4: Understanding Post-installation Tasks **55**

How You Further Customize Your Instance.....	55
How You Prepare Your Installation for General Use.....	56
How You Safely Uninstall Ingres	57
Uninstall Ingres for Linux Using the Installation Wizard	57
uninstall_ingres Command—Uninstall Ingres for Linux.....	58
Uninstall Ingres for Windows	58

Appendix A: Installing Ingres on UNIX **59**

How You Prepare for Installation on UNIX	59
How You Create the System Administrator Account on UNIX	60
Major Configuration Options	61
How You Install and Configure Ingres in a UNIX Environment	62
Set TERM_INGRES	63
Installation Modes.....	63
Extract Files from the Distribution	64
ingres_express_install Command—Install Ingres Quickly.....	64
Install Ingres in Interactive Mode	65
Install Ingres in Command Line Mode	65
Set II_DISTRIBUTION	66

Ingbuild Command–Install Ingres	67
Run the Setup Programs	70
How You Install and Configure NFS Clients.....	72
Set Up Each NFS Client Individually.....	72
Set Up NFS Clients from the DBMS Server.....	74
Post-Installation Tasks.....	75
Transaction Log File as a Raw Device.....	75
Automatic Startup Command	78
Allow Access on Systems Using Shadow Passwords	79
How You Establish User Access to Tools and Databases	79
View Contents of Your Current Installation	81

Appendix B: Installing Ingres on OpenVMS **83**

How You Install and Configure Ingres (First Time Installers).....	83
How You Install and Configure Ingres (Experienced Installers).....	84
Define the TERM_INGRES Logical	85
Major Configuration Options	86
Multiple Instances on One Node	87
File Locations and Logicals.....	88
Define II_SYSTEM as a Logical.....	88
Define a Concealed, Rooted Logical	89
Guidelines for VMS Cluster Configuration.....	89
Owner of the System Administrator Account	90
UIC for an Installation	90
Run the Setup Programs	91
Post-Installation Tasks.....	92
How You Add an Automatic Startup Command	93
How You Establish User Access to Tools and Databases	93
Installation History Log File	94
Ingres Cluster Solution for OpenVMS	96
Requirements for the Ingres Cluster Solution on OpenVMS	96
How You Prepare to Install the Ingres Cluster Solution for OpenVMS	97
How You Install the Ingres Cluster Solution for OpenVMS	98
How Client Applications Access an Ingres Cluster	99

Appendix C: Configuring the High Availability Option for Red Hat Cluster Suite **101**

What You Need to Know.....	101
A Database Service	101
How You Set Up a Cluster Service for an Ingres Database.....	102
Add an Ingres Service.....	104

Appendix D: Configuring the High Availability Option for Windows Clustering **105**

What You Need to Know.....	105
Ingres High Availability Option	106
Failover	106
Configuration Guidelines for System and Data Files.....	106
Cluster Disk.....	106
Physical Disk Resource Type.....	107
Generic Service Resource Type.....	107
DBMS Server Requirements.....	107
Virtual Servers	108
Virtual Server Name.....	108
Windows Cluster Service Installation.....	108
Cluster Administrator	108
Verify Windows Cluster Service Installation	108
Cluster Command	109
Prerequisites for High Availability Option	109
Install High Availability Option for Windows Clustering.....	110
Functions Performed by the Ingres High Availability Option Setup Utility.....	110
Installation on Windows Server 2008.....	110
Starting and Stopping Ingres in a Cluster Environment.....	111
Start Ingres in a Cluster Environment.....	111
Stop Ingres in a Cluster Environment	112
How the High Availability Option Is Removed	113

Appendix E: Configuring the High Availability Option for Sun **115**

What You Need to Know.....	115
Failover	116
Configuration Guidelines for System and Data Files.....	116
Cluster File Systems	116
DBMS Server Requirements.....	116
Sun Cluster Software Installation Verification	117
How You Install the Ingres High Availability Option for Sun	117
How to Respond to Installation Script Prompts.....	118
How the Ingres High Availability Option Is Configured	120
How the Database Probe Function Is Configured	121
Turn on Debugging and Tracing for the High Availability Option	122
Starting and Stopping Ingres in a Cluster Environment.....	122
Start Ingres in a Cluster Environment.....	122
Stop Ingres in a Cluster Environment	123
Clear a Fault in a Resource Group.....	123

How You Remove the High Availability Option	124
---	-----

Appendix F: System Requirements for OpenVMS **125**

System Hardware Requirements on OpenVMS	125
SYSGEN Parameters that Affect Operation	126
Recommended OpenVMS Process Resources	129
OpenVMS Quotas and Ingres Parameters	130
Required OpenVMS Privileges.....	131

Appendix G: Response File Parameters **133**

II_RESPONSE_FILE	133
Response File—Define Configuration for Ingres Instance	133
List of Response File Parameters	134
Example Response File for Linux	136
Example Response File for Windows.....	136
Response File Parameters.....	138
II_ADD_REMOVE_PROGRAMS	138
II_ADD_TO_PATH	138
II_CHARSET	139
II_CHECKPOINT	139
II_COMPONENT_CORE	139
II_COMPONENT_DBMS	140
II_COMPONENT_DOCUMENTATION	140
II_COMPONENT_DOTNET.....	140
II_COMPONENT_FRONTTOOLS	141
II_COMPONENT_JDBC_CLIENT	141
II_COMPONENT_NET.....	142
II_COMPONENT_ODBC	142
II_COMPONENT_REPLICATOR.....	143
II_COMPONENT_SPATIAL	143
II_COMPONENT_STAR.....	143
II_COMPONENT_VISION	144
II_CONFIG_TYPE.....	144
II_DATABASE.....	144
II_DATE_TYPE_ALIAS.....	145
II_DATE_FORMAT	145
II_DEMODB_CREATE.....	146
II_DESTROY_TXLOG	146
II_DUAL_LOG	147
II_DUMP	147
II_ENABLE_NETBIOS	147

II_ENABLE_SPX.....	148
II_ENABLE_SQL92	148
II_ENABLE_TCPIP	148
II_GROUPID	149
II_INSTALL_ALL_ICONS	149
II_INSTALLATION	149
II_JOURNAL.....	150
II_LOCATION_DOCUMENTATION	150
II_LOCATION_DOTNET	150
II_LOG_FILE	150
II_LOG_FILE_SIZE_MB.....	151
II_MONEY_FORMAT	151
II_REMOVE_ALL_FILES.....	152
II_SERVICE_START_AUTO	152
II_SERVICE_START_USER	152
II_SERVICE_START_USERPASSWORD	153
II_START_INGRES_ON_COMPLETE.....	153
II_START_IVM_ON_COMPLETE	153
II_START_ON_BOOT	154
II_SYSTEM	154
II_TERMINAL	154
II_TIMEZONE_NAME	156
II_UPGRADE_USER_DB	156
II_USERID.....	156
II_WORK.....	156

Chapter 1: Understanding Installation Considerations

This section contains the following topics:

[An Ingres Instance](#) (see page 9)

[System Administrator Account](#) (see page 10)

[Ingres Files](#) (see page 11)

[Location of Ingres Files](#) (see page 12)

[File Location Guidelines](#) (see page 13)

[Use of Ingres Locations](#) (see page 17)

[Sample Disk Configurations](#) (see page 22)

[World Region and Time Zone](#) (see page 24)

[Character Set](#) (see page 26)

[Date Alias Setting](#) (see page 28)

[Installation Parameters](#) (see page 28)

[Ingres Servers](#) (see page 31)

[Valid Computer, Directory, and User Names](#) (see page 32)

An Ingres Instance

An Ingres instance consists of a set of installed products that share a unique system-file location, ownership, and instance name, together with any data files created by these products.

Multiple copies of Ingres can be installed on a single server and run simultaneously. Each copy of Ingres is referred to as an Ingres instance.

An instance is classified as either a server installation or a client installation.

The default instance name is: Ingres II.

Instance ID

The instance ID is a two-character code that identifies a specific instance on a node and allows all processes and images to be installed and shared successfully. The value of the instance ID is stored in the `II_INSTALLATION` environment variable.

In the default instance name Ingres II, the instance ID is II.

The first character of an instance ID must be a letter; the second character can be a letter or numeral. The default instance ID is II.

Multiple Instances

If you have more than one instance on the same node, each instance on that node must have a unique instance ID. For example, you can install and run a new version of Ingres under one instance name, while maintaining an existing older version under a different instance name on the same computer or network node.

System Administrator Account

The system administrator account is a user account that is used for system management on each Ingres instance. The system administrator account owns the instance, and so the system administrator is often referred to as the instance owner.

This account is created when installing the DBMS Server. Ingres must be installed under this account because every DBMS system file requires ownership by this user.

Note: Use the system administrator account only for Ingres system management work. We recommend that you do not put non-Ingres files in this account.

The system administrator must have privileges to:

- Perform system administration functions (such as starting and stopping servers)
- Create databases
- Establish other users and grant privileges to them

Ingres Files

An Ingres instance includes the following files:

- System files (executables)
- Database files

These files include the master database and user databases. The master database, `iidbdb`, stores information about all databases, their locations, and the users who can access them.

- Transaction log file

This file stores uncommitted transactions and buffers committed transactions before they are written to the database. Ingres uses one logical instance-wide log file. The file is circular and wraps when it encounters the physical end-of-file.

Each logical log file may consist of up to 16 physical disk files, referred to as partitions, which helps to alleviate I/O bottlenecks. To ensure that no committed transactions are lost if the primary transaction log devices fail, Ingres can maintain a backup of the primary transaction log file, also known as a dual log, on the storage locations you specify.

- Checkpoint files, journal files, dump files

Checkpoint, journal, and dump files provide for data recovery in case of a database disk failure.

Checkpoints alone provide for data recovery up to the time of the checkpoint.

Checkpoints and journals provide for recovery up to the time of failure.

Checkpoint, journal, and dump files provide online checkpoints.

- Temporary work files

Work files are temporary files created during external sorts and other DBMS Server operations that require large amounts of temporary file space.

Location of Ingres Files

Before installing Ingres, you must decide on locations for the Ingres files. Choose these locations carefully because they cannot be easily changed once specified. The exception to this rule is the location of the transaction log.

During installation, you will provide the directory path for each instance default location, and it will be recorded in an Ingres environment variable/logical (for example, the location for the system files is stored in `II_SYSTEM`).

After installation, you can create additional locations of all types except `II_SYSTEM`. Every user database must have at least one Database, Work, Checkpoint, Dump, and Journal location, but these locations can be anywhere and no database—except the master (`iidbdb`)—is required to use the instance default locations. If you have enough disks to create additional locations, the choice of locations for each database should follow the same guidelines as for the instance defaults.

For development or test machines where you periodically refresh a copy of the production database, we recommend that the two copies of the database be "checkpoint compatible" (same Data, Work, Checkpoint, Dump, and Journal location names and paths). Checkpoint compatibility allows simple restoration of a production checkpoint to the test instance.

File Location Guidelines

You should consider the following factors when choosing the locations for the Ingres files:

- Ability to recover from single component (disk, volume, file system) failure

If a single component fails, a well-designed system is easily recoverable. For example, if a component failure renders the online system unusable, you should be able to replace the failed component and recover the system with no loss of data. Beware of locating stripes from different volumes on one disk—if the disk breaks, both volumes are lost.

You should consider the cost of downtime and the time required for various recovery scenarios. When downtime is expensive, you might consider more capable and reliable storage technology and extra redundancy.

- Sufficient I/O bandwidth for each location

Under all modes of operation, the system should not have insufficient reading or writing bandwidth through one channel and excess bandwidth through another. If you need more bandwidth, you can stripe a volume across more disks. Do not provide excess bandwidth to locations that do not need it.

- Sufficient disk space for each location

Disk space requirements for some locations can only be determined by you, and depend on factors such as how large your database is.

II_SYSTEM (System Files) Location Guidelines

The location you choose for Ingres system (executable and script) files will also contain the error log and configuration files. Choose a location that has adequate disk space (at least 2 GB free).

If you lose this location, Ingres will fail. Replace the failed component, restore the file system from backup. Ingres will recover on startup.

If you fill this file system, Ingres will fail. Free some space or allocate additional space, and restart Ingres if necessary.

The I/O bandwidth requirement for II_SYSTEM is low.

II_DATABASE (Database Files) Location Guidelines

The instance default data location contains the master database (iibddb), which stores information about all databases, their locations, and the users that can access them. By default, this location also contains all user databases, unless the database administrator specifies an alternate location for a database when creating it.

When choosing the data location, consider the following:

- Place database files on a separate disk from checkpoint, journal, and dump files to maximize chances for data recovery.
- Place database files on a separate disk from the transaction log files to improve system performance by distributing disk I/O.
- Place database files on a separate disk from the work locations. Doing so prevents work operations that might fill the file system from having an impact on the data location, reduces fragmentation of both locations, and ensures sufficient reading bandwidth for the data location.
- On systems with three or more disks, do not place the database files on the same disk as your operating system. Administrators typically will want to improve reliability by avoiding a large amount of file create, open, and close activity on the operating system disk and not risk the operating system disk running out of space.
- Keep the usage of data file systems below 75 percent. Above this level, fragmentation will be more rapid and will degrade the efficiency of scanning operations. You will also need sufficient space to reorganize tables as part of database maintenance.

If you lose this location, Ingres will fail. You must replace the failed component, recreate the directory structure, and recover from checkpoint and journals all Ingres databases that use this location.

If you fill this file system, Ingres will fail. Free some space or allocate additional space, and restart Ingres if necessary.

The I/O bandwidth requirement for this file system depends on your reading and writing rates to the user databases using it (iibddb requires almost none). Unless these are well-tuned OLTP databases, partial and complete scans of tables will dominate the reading. A larger Ingres cache will reduce reading, but will have almost no effect on substantial table scanning.

Warnings! We strongly recommend that you do not use the following for data locations:

- ReiserFS due to severe performance degradation noticed when using this file system

- A RAID5 storage configuration—Striping increases performance; redundancy increases data security. Economizing on disks will typically not pay in the long run.

II_CHECKPOINT, II_JOURNAL, II_DUMP (Checkpoint, Journal, Dump Files) Location Guidelines

Checkpoint files, journal files, and dump files can reside on the same device because journals and dump files are useful in recovery only if the associated checkpoint is also available. By default, the install program places journal and dump files in the same location as the checkpoint files.

Important! Do not place the Checkpoint location on the same disk as any location used by the online system (except II_DUMP). Losing the online system and the database backup in the same failure would require a more complex and longer recovery from offline media. On single-disk systems, we recommend checkpointing to magnetic tape.

If you lose the Dump location or fill the file system, Ingres will fail as it attempts to update the "dump copy" of each database configuration file. You must recreate these directories, copy each database configuration file from the database root data directory and restart Ingres.

If you lose the Journal location, any journaled databases using it will cause the archiver to shut down. You then have between a minute and an hour (before the transaction log file fills up) to disable journaling on each affected database. Then you must replace the failed component and rebuild the directory structure. To restart journaling you must run an offline checkpoint on each database. If you fill the journal file system, free disk space and restart the archiver.

If you lose the Checkpoint location, the online system is not affected. However, you now have no local backup for the databases using this location. Replace the failed component, rebuild the directory structure, and hope that you do not need a checkpoint until the next one is created. If you fill the checkpoint file system, the checkpoint process will fail. You must free sufficient space before the next checkpoint.

The Dump location requires negligible bandwidth. The Journal location requires a moderate amount of writing bandwidth during each archiver run. The Checkpoint location requires an enormous amount of writing bandwidth as each checkpoint is run.

II_WORK (Temporary Work Files) Location Guidelines

If your system will run a significant number of queries involving large amounts of data, then you will need a lot of reading and writing bandwidth in the work location.

If you lose this location, replace the failed component, rebuild the directory structure, and restart Ingres. No files need to be recovered.

If you fill this file system, Ingres will usually recover automatically as the query fails and the offending work file is removed.

For more information about temporary files and sorting, see the *Database Administrator Guide*.

Transaction (and Backup) Log Location Guidelines

You can optionally configure a backup transaction log file for use. The online system will write identical transactions to both copies. This dual logging can limit performance during heavy log writing. Volume mirroring is faster, but loss of both sides of the mirror has been known.

Consider the following when choosing the locations for the transaction or backup transaction log file:

- Do not locate the primary and backup transaction log files on the same disk.
- If possible, avoid installing the primary or backup transaction log files on the same disk as your data location because long (5 minutes or longer) peaks in writing activity will stress both the data and transaction log file writing bandwidth.
- Put the transaction log file on separate disks from your checkpoint, dump, and journal files so that you can perform offline recovery of databases if the transaction log file disk fails.

Note: On Ingres for Linux, the backup transaction log is disabled by default. To enable the backup transaction log, set the environment variable `II_DUAL_LOG` to the location of the backup log file (either in the response file or prior to installing).

On UNIX (but not Linux) the transaction log file can be located on a raw partition to increase writing bandwidth. Bypassing the file system also eliminates the small possibility of loss through file system failure.

Use of Ingres Locations

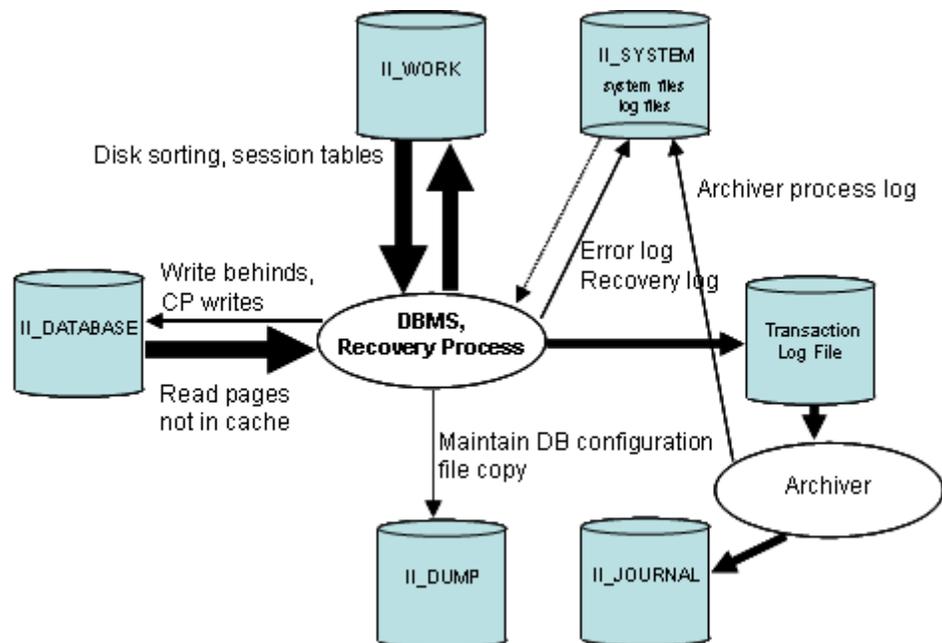
You can plan better for your installation if you understand how Ingres locations are typically used by various Ingres tasks.

An Ingres instance can perform several different types of tasks: normal operations, database maintenance, checkpoint, recovery. Some of these may proceed concurrently. Each task requires access to some of the Ingres locations.

For a simple instance using only the default locations, the following sections give an overview of which locations are used during which operations and the likely bandwidth requirements.

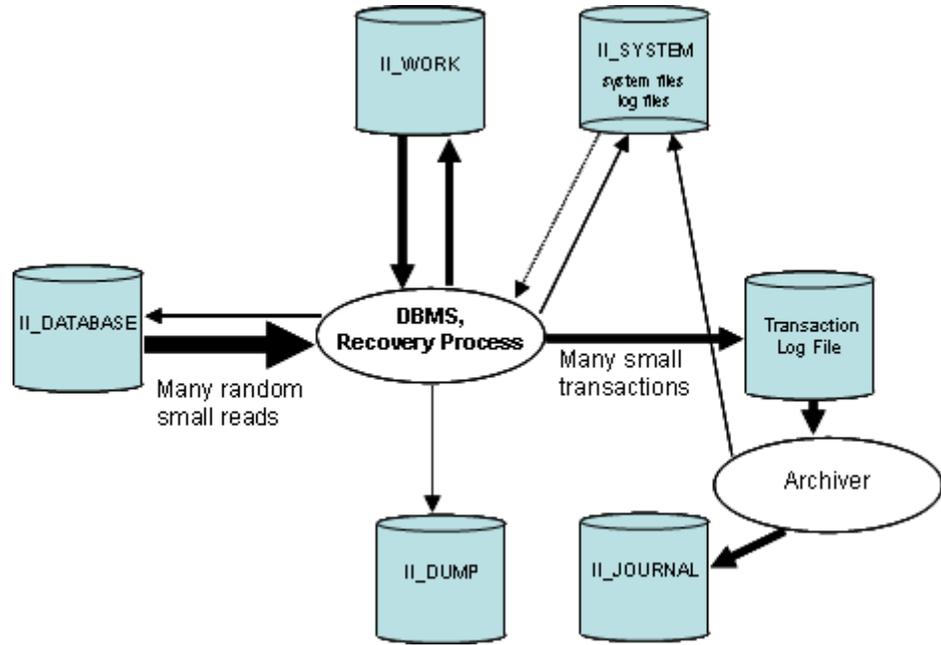
Location Use During Normal Operations

During normal operations, all locations except II_CHECKPOINT must be accessible. The busiest channels are typically those reading from the Data location and writing to the Transaction Log File. Heavier updating will increase writing to the Transaction Log File and (asynchronously) to the Data location. At a configurable interval, the Archiver reads everything written to the Transaction Log File and writes the journaled records to the Journal location.



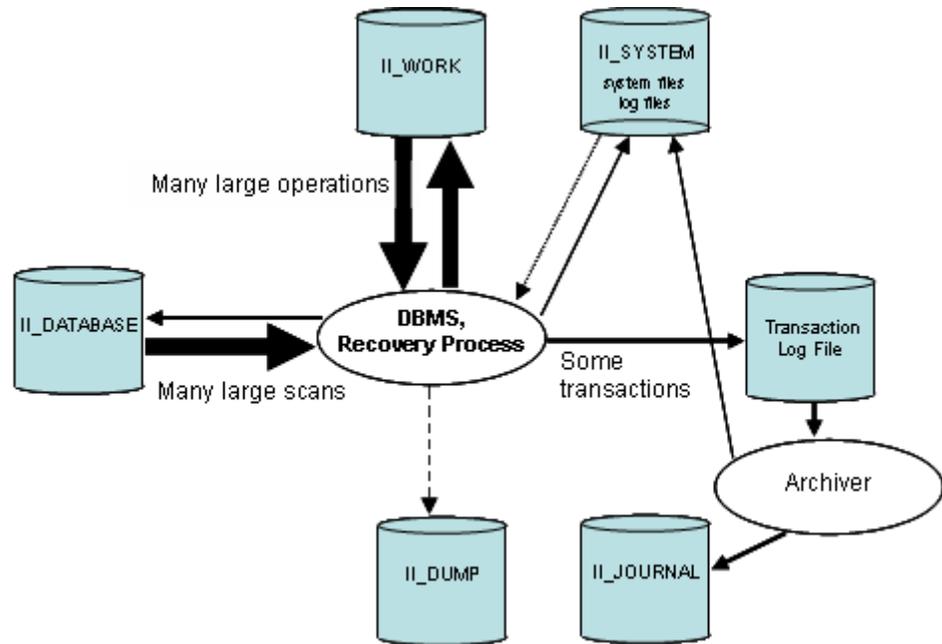
Location Use During OLTP Operations

If your system processes small transactions and no queries handle large amounts of data—as in an Online Transaction Processing system—then the use of all locations is more uniform and predictable. There will be many small random reads instead of scanning of data files. The Work location will be used less because small sorts occur in memory.



Location Use During DSS or OLAP Operations

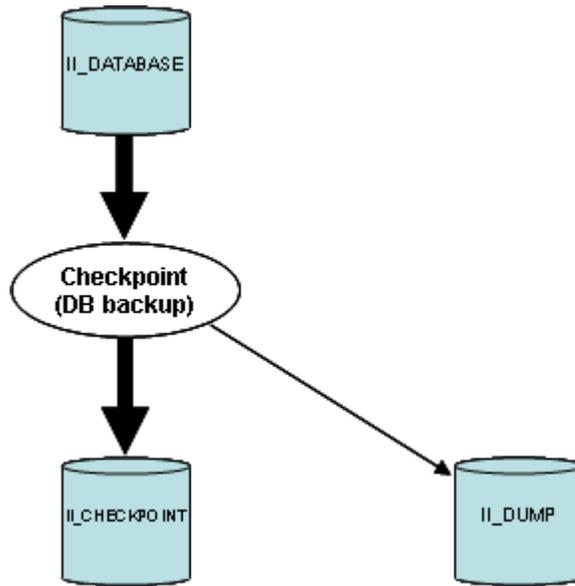
If your system queries and aggregates large volumes of data—as in a Decision Support System or Online Analytical Processing system—reading from the Data location occurs more in the form of partial and complete file scans. Manipulation of large data sets occurs in the Work location. Updating, and therefore logging, in such a system is typically minor.



Location Use During Checkpointing

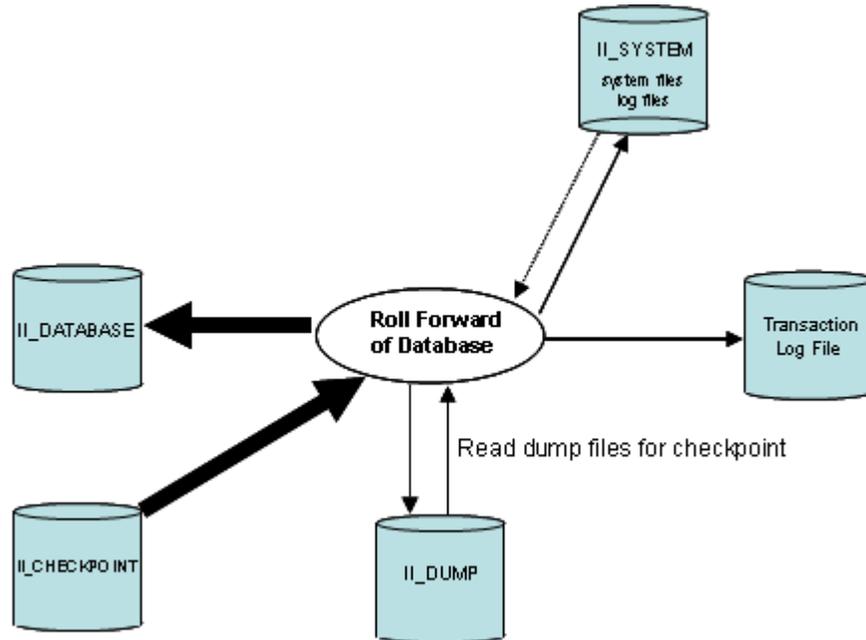
A checkpoint (database backup) reads all the data from the Data location and writes it to the Checkpoint location. Checkpointing requires the most reading for the Data location, and is the only time Ingres writes to the Checkpoint location. Any updates during the checkpoint result in writes to the Dump location.

Dynamically compressing checkpoints is a common practice and decreases the disk space and writing bandwidth requirement to about one-tenth, while using more CPU.



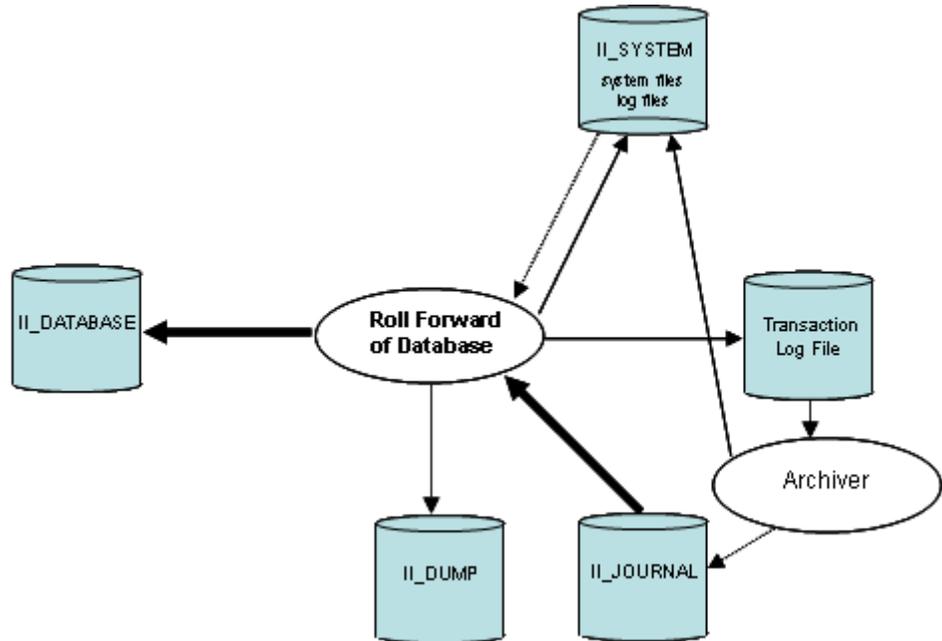
Location Use During Recovery from a Checkpoint

Recovery from a checkpoint is the opposite process used during checkpointing. Ingres reads the backup files from the Checkpoint location and writes them back to the Data location.



Location Use During Replay of Journals

Journal replay reads from the Journal location and writes to the Data location. Associated administrative tasks require access to other locations.



Sample Disk Configurations

This section shows typical disk configurations for an Ingres installation. These sample configurations illustrate the recommended file location guidelines. Where more sophisticated storage options are available, the same principles apply.

Four-Disk DBMS Server Configuration

A configuration with four or more disks provides the best performance and recovery options. For best performance, configure your system with the operating system on a separate disk and your Ingres files on three or more other disks, as follows:

Disk 1—Operating system files

Disk 2—II_CHECKPOINT, II_JOURNAL, and II_DUMP

Disk 3—II_SYSTEM, II_WORK, and transaction log

Disk 4—II_DATABASE

If possible, put your backup transaction log on a separate disk from both your database and primary transaction log. Also, you can use more than one disk partition for your transaction log.

Three-Disk DBMS Server Configuration

The following three-disk configuration provides better performance and additional recovery options than a two-disk system:

Disk 1—Operating system, II_CHECKPOINT, II_JOURNAL, and II_DUMP

Disk 2—II_SYSTEM, II_WORK, and transaction log

Disk 3—II_DATABASE

Your backup transaction log can reside on either Disk 1 or Disk 3.

Two-Disk DBMS Server Configuration

A two-disk system is the minimal recommended configuration because certain types of recovery are still possible:

Disk 1—II_DATABASE, optional backup transaction log file

Disk 2—All other locations

One-Disk DBMS Server Configuration

A single-disk system is a high-risk, low-performance setup and not recommended. If the disk fails, you could lose all your data. On single-disk systems, you should checkpoint to remote storage such as an NFS-mounted file system.

Note: The default installation method installs Ingres as a one-disk system.

Client-only Disk Configuration

In the Client configuration, the Ingres Net and Ingres tools system files reside on a single disk on the client instance. There are no server components in a client-only install.

NFS Client Disk Configuration (UNIX)

In the Network File System (NFS) configuration, no files reside on the NFS client instance. All Ingres files reside on the networked DBMS Server instance.

The NFS client instance uses the Network File System to share executables and other Ingres files. The `ingmknfs` utility, which you use to create and configure NFS clients, creates an NFS admin directory on the DBMS Server instance to store NFS client-related files.

World Region and Time Zone

You must specify the time zone for your instance. This value is stored in the `II_TIMEZONE_NAME` environment variable.

On some systems, the default value for `II_TIMEZONE_NAME` is `NA-PACIFIC`. If you are in a different time zone, you must change the value of `II_TIMEZONE_NAME`.

Time zone names are organized by world region. In some cases, the time zone name is a positive or negative offset from Greenwich Mean Time (for example, `GMT2` or `GMT-2`). If you are unable to locate the correct time zone within one of the designated world regions, use the `GMT-OFFSET` world region and specify one of the GMT offsets as your time zone.

The time zone parameter tells Ingres what adjustments to make for Daylight Savings Time. If you must make other adjustments for special time changes imposed in your area (such as for energy conservation purposes), you can use the `iizic` time zone compiler provided in the distribution.

Time Zone Names

The world regions and their time zone names are as follows:

Africa

GMT
 GMT1
 GMT2
 GMT3
 GMT4

Asia

INDIA
 JAPAN
 KOREA
 HONG-KONG
 PAKISTAN
 PRC
 ROC
 GMT5
 GMT6
 GMT7
 GMT8
 GMT9
 GMT10
 GMT11

Australia

AUSTRALIA-LHI
 AUSTRALIA-NORTH
 AUSTRALIA-WEST
 AUSTRALIA-SOUTH
 AUSTRALIA-TASMANIA
 AUSTRALIA-QUEENSLAND
 AUSTRALIA-VICTORIA
 AUSTRALIA-NSW
 AUSTRALIA-YANCO

Middle East

EGYPT
 IRAN
 ISRAEL
 KUWAIT
 SAUDI-ARABIA
 GMT2
 GMT3
 GMT4

North America

NA-PACIFIC
 NA-MOUNTAIN
 NA-CENTRAL
 NA-EASTERN
 NA-ALASKA
 CANADA-ATLANTIC
 CANADA-NEWFOUNDLAND
 CANADA-YUKON
 MEXICO-GENERAL
 MEXICO-BAJANORTE
 MEXICO-BAJASUR

North-Atlantic

EUROPE-WESTERN
 EUROPE-CENTRAL
 EUROPE-EASTERN
 IRELAND
 MOSCOW
 POLAND
 TURKEY
 UNITED-KINGDOM
 GMT
 GMT1
 GMT2
 GMT3

South America

BRAZIL-EAST
 BRAZIL-WEST
 BRAZIL-ACRE
 BRAZIL-DENORONHA
 CHILE-CONTINENTAL
 CHILE-EASTER-ISLAND
 GMT6
 GMT5
 GMT4
 GMT3

South Pacific

NEW-ZEALAND
 US-HAWAII
 GMT10
 GMT11
 GMT12
 GMT-12
 GMT-11
 GMT-10

Southeast Asia

INDONESIA-WEST
 INDONESIA-CENTRAL
 INDONESIA-EAST
 MALAYSIA
 PHILIPPINES
 SINGAPORE
 THAILAND
 VIETNAM
 GMT7
 GMT8
 GMT9

GMT-Offset

GMT-12
 GMT-11
 GMT-10
 GMT-9
 GMT-9-and-half
 GMT-8
 GMT-7
 GMT-6
 GMT-5
 GMT-5-and-half
 GMT-4
 GMT-3
 GMT-3-and-half
 GMT-2
 GMT-2-and-half
 GMT-1
 GMT
 GMT1
 GMT2
 GMT3
 GMT3-and-half
 GMT4
 GMT5
 GMT5-and-half
 GMT6
 GMT7
 GMT8
 GMT9
 GMT9
 GMT10
 GMT10
 GMT11
 GMT12
 GMT13

Character Set

You must select the character set during installation. The installation program provides a default value.

Important! After Ingres is installed, you cannot change the character set from its current setting (II_CHARSET) at any time without risking the corruption of your data.

Ingres-supported character sets are as follows:

Character Set	Description	Format
ALT	Support of Cyrillic on DOS	Single byte
ARABIC	Arabic-449-Plus	Single byte
CHINESES	Simplified Chinese - PRC	Double byte
CHINESET	Traditional Chinese - Taiwan	Double byte
CHTBIG5	Traditional Chinese - Taiwan, BIG5	Double byte
CHTEUC	Traditional Chinese - Taiwan, EUC	Double byte
CHTHP	Traditional Chinese - Taiwan, HP ROC15	Double byte
CSGB2312	Simplified Chinese - GB2312	Double byte
CSGBK	Simplified Chinese - GBK	Double byte
CW	Cyrillic on Windows 3.1	Single byte
DECMULTI	DEC Multinational (superset of ASCII) and default for VMS	Single byte
DOSASMO	IBM DOS ASMO Arabic (cp708)	Single byte
ELOT437	Greek for PC/RS6000/SCO-UNIX	Single byte
GREEK	DEC Greek Elot	Single byte
HEBREW	DEC Hebrew	Single byte
HPROMAN8	HP Roman8 (superset of ASCII)	Single byte
IBMPC437	IBM PC Code Page 437 (US and English)	Single byte
IBMPC850	IBM PC Code Page 850 (Multilingual), includes accented characters	Single byte
IBMPC866	IBM PC 866 (Cyrillic for DOS)	Single byte
IS885915	ISO 8859/2 (Latin and some Greek). Identical to ISO 8859/1 Latin, except for eight characters, including the Euro currency	Single byte

Character Set	Description	Format
	symbol (€, Unicode U+20AC).	
ISO88591	ISO 8859/1 Latin and default for UNIX (superset of ASCII)	Single byte
ISO88592	8859/5 (Latin and Cyrillic)	Single byte
ISO88595	8859/9 (Latin and some Turkish) CP 920	Single byte
ISO88597	ISO 8859/7 (Greek)	Single byte
ISO88599	ISO 8859/15 (Latin and Euro sign)	Single byte
KANJIEUC	Japanese, EUC	Double byte
KOI18	KOI 8-bit (ISO 6937/8), Russia	Single byte
KOREAN	Korean	Double byte
PC737	IBM PC Code page 737 - Greek	Single byte
PC857	IBM PC Code page 857 - Turkish	Single byte
PCHEBREW	IBM PC / MSDOS Hebrew	Single byte
SHIFTJIS	Shift-JIS Japanese	Double byte
SLAV852	IBM PC Code Page 852 (Slavic)	Single byte
THAI	DEC Thai Tis	Single byte
UTF8	Unicode encoding form UTF-8	Multi-byte
WARABIC	Arabic	Single byte
WHEBREW	Microsoft Windows Hebrew	Single byte
WIN1250	Eastern Europe: Windows page 1250	Single byte
WIN1252	Windows code page 1252 - Latin 1 (Western Europe) and default for Windows	Single byte
WIN1253	Modern Greek	Single byte
WTHAI	IBM/Windows Thai (cp874)	Single byte

Note: On Linux, the character set value must be entered in uppercase.

Note: ja_JP.UTF-8 on Japanese SUSE Linux is not supported.

Date Alias Setting

The `date_alias` configuration parameter controls whether the keyword `DATE` used to define a column data type refers to the `ANSIDATE` or `INGRESDATE` data type. During installation, this parameter is assigned a value by default. The setting can be changed during a custom installation.

The setting can also be changed after installation. The `date_alias` parameter, which is not available in the configuration utilities, must be set with the `iisetres` command, as follows:

```
iisetres ii.machine.config.date_alias value
```

where *machine* is the machine name, and *value* is either `ANSIDATE` or `INGRESDATE`.

Installation Parameters

During the installation process, values are assigned for several installation and configuration parameters. The values are stored in environment variables.

General Installation Parameters

General installation parameters apply to all instances.

The general installation parameters are as follows (environment variables are shown in parentheses):

Instance ID (II_INSTALLATION)

Identifies the instance, and is part of the instance name.

Installation location (II_SYSTEM)

Defines location of the product's system (executable) files.

Instance Owner (II_USERID) (Linux and UNIX only)

Defines the user ID of the system administrator that owns the Ingres instance. The ID is added to the system if it does not exist.

Group ID (II_GROUPID) (Linux and UNIX only)

Defines the group ID to which the system administrator user ID belongs and that owns the Ingres instance. The ID is added to the system if it does not exist.

Silent install process (II_SILENT_INSTALL) (Linux only)

Quiets installation messages except those from RPM.

Start instance with OS (II_START_ON_BOOT) (Linux only)

Controls whether the instance will be started automatically when the operating system boots.

Start Ingres Automatically at System Startup (II_SERVICE_START_AUTO) (Windows only)

Controls whether the instance will be started automatically when the operating system boots.

Start Ingres under....account (II_SERVICE_START_USER) (Windows only)

Specifies the user ID that starts Ingres as a service.

DBMS Server Installation Parameters

The parameter settings when installing the Ingres DBMS Server are as follows (environment variables are shown in parentheses):

Database location (II_DATABASE)

Defines the location for the Ingres master database (iiddb) and the default location for database files.

Checkpoint files location (II_CHECKPOINT)

Defines the location for the checkpoint files that serve as a static backup of the database.

Journal files location (II_JOURNAL)

Defines the location for the journal files, which provide a dynamic record of changes made to Ingres databases since the last checkpoint.

Dump files location (II_DUMP)

Defines the location for the dump files used to perform online backups.

Temporary work files location (II_WORK)

Defines the location for temporary files created during external sorts and other DBMS Server operations.

Transaction log location (II_LOG_FILE)

Defines the location of the transaction log file.

Dual transaction log location (II_DUAL_LOG)

Defines the location of the backup transaction log file.

Transaction log file size (II_LOG_FILE_SIZE_MB)

Defines the default size of the transaction log file. The default size (256 MB per file) is adequate for most instances. You should change the file size only if you have an existing application that requires a larger transaction log file.

Connection Limit (II_CONNECT_LIMIT)

Specifies the maximum number of concurrent users.

Time zone (II_TIMEZONE_NAME)

Specifies the region and time zone.

Character set (II_CHARSET)

Specifies the character set.

Date Alias (II_DATE_TYPE_ALIAS)

Specifies whether date columns should use ANSI date or Ingres date data type.

Configuration type (II_CONFIG_TYPE)

Specifies a configuration type, which determines Ingres configuration parameter settings based on how Ingres will be used.

Ingres Servers

The following server types are available, depending on your particular instance:

Bridge Server

Enables network communications between an Ingres client using one network protocol and an Ingres server using a different network protocol.

Communications Server

Enables network communications between an Ingres client and an Ingres server instance on a remote machine through Ingres Net.

DBMS Server

Allows users access to your instance's databases.

Data Access Server

Translates requests from the Ingres JDBC Driver and .NET Data Provider and forwards them to the appropriate DBMS Server.

Name Server

Keeps track of all the servers associated with an instance.

Recovery Server

Handles online recovery from server and system failures.

Remote Command Server

Enables the execution of Ingres commands on a remote Ingres instance. This server is used by the Visual DBA suite of tools.

Star Server

Allows users to connect to multiple databases simultaneously through Ingres Star.

Valid Computer, Directory, and User Names

Computer, directory, and user names must be valid for the operating system that Ingres is being installed on and also adhere to the following restrictions.

Computer names

Ingres supports computer names that contain the following ASCII characters only: alphanumeric and hyphen (-). Computer names must not start with a hyphen.

Directory path

A valid Ingres directory path has slightly different syntax on UNIX and Windows.

On both UNIX and Windows, a directory name must start with an ASCII letter (that is, a-z and A-Z) or a digit (0-9), and can continue with ASCII letters, digits, and the following symbols: space, hyphen (-), underscore (_), full stop (.), colon (:), left square bracket ([), right square bracket (]), tilde (~).

Note: Windows does not allow a colon in a directory name.

UNIX and Linux: Absolute directory paths must start with a forward slash (/) and continue with a relative directory path. A relative directory path contains a number of directory names separated by /, as in the following example:

```
/usr/local/ingresII
```

Windows: Absolute directory paths must start with a drive specification (a drive letter immediately followed by a colon), a backward slash (\), and then a relative directory path, as in the following example:

```
c:\program files\ingres\ingresII
```

The directory name can also start with underscore (_).

Note: On UNIX and Linux, uppercase letters are distinct from lowercase ones (for example, a is not the same as A). On Windows, uppercase letters are not distinct from lowercase ones (for example, a is the same as A). In general, we recommend that path names not be distinguished by case.

User names

Ingres supports user names that contain the following ASCII characters only: alphanumeric, at (@), pound (#), dollar (\$), underscore (_). User names must not start with @, #, \$, or a digit. These restrictions apply to the user names used to install Ingres and to connect to Ingres. Passwords must not exceed 20 characters.

Chapter 2: Installing Ingres

This section contains the following topics:

[How You Install Ingres](#) (see page 33)

[Installing Ingres for Linux](#) (see page 33)

[Install Ingres for Windows](#) (see page 41)

[Install Ingres for UNIX](#) (see page 44)

[Install Ingres for OpenVMS](#) (see page 44)

[How You Access the Instance on Linux and UNIX](#) (see page 46)

[Configuration Type Parameter Settings](#) (see page 47)

How You Install Ingres

If you are installing Ingres for the first time, you should be familiar with the Installation Considerations (see page 9) before installing the software.

The process for installing Ingres in any operating system environment is as follows:

1. Verify that your hardware and software meet the Ingres requirements, as documented in the Readme for your platform.
2. Decide on file locations, as described in File Location Guidelines (see page 13) and Sample Disk Configurations (see page 22).
3. Install the software.
4. Perform post-installation tasks (see page 55).

Installing Ingres for Linux

Ingres is installed on Linux using RPM Package Manager. You can install Ingres using an installation wizard or from the command line.

Note: Installation of Ingres requires root access. If you are logged in as another user, you will need the root password.

Note: If your system does not have RPM, Ingres for Linux is also distributed in a format that lets you use the ingbuild installation program. Follow the instructions described in "Installing Ingres on UNIX" in the *Installation Guide*.

Install Ingres for Linux Using the Installation Wizard

The installation wizard easily installs Ingres without your having to know RPM commands.

To start the Ingres Installation Wizard

1. Run the `ingres_install` script located in the root directory of the Ingres distribution.

The first page of the Ingres Installation Wizard is displayed.

2. Respond to the installation wizard dialogs.

If you need more information about a specific item, click the item's information button .

After you provide the required information in the installation wizard dialogs, the installation program installs Ingres.

Installing Ingres for Linux at the Command Line

You can install the Ingres RPM packages at the command line by doing either of the following:

- Using the `ingres_express_install` command
- Using RPM commands

`ingres_express_install` Command—Install Ingres

The `ingres_express_install` command quickly installs Ingres RPM packages.

This command has the following format:

```
ingres_express_install [-tar] [instance_PATH] [instance_ID]
```

-tar

Installs from `ingres.tar` instead of RPMs. (Linux only)

instance_PATH

Identifies the full path to the location where Ingres is to be installed (II_SYSTEM).

Default: `/opt/Ingres/IngresII`

instance_ID

Defines a two-character string where the first character must be an uppercase letter and the second character must be an uppercase letter or a number from 0 to 9.

Default: `II`

Notes:

1. The `ingres_express_install` command must be run as the "root" system user.
2. An "ingres" system user must exist; otherwise `ingres_express_install` fails.
3. The "ingres" system user must have read, write, and execute permissions on the `II_SYSTEM` directory; otherwise `ingres_express_install` fails.
4. If an alternate *instance ID* (for example, A2) is specified and the *instance PATH* is not specified, `ingres_express_install` tries to install into `/opt/Ingres/IngresII`, which can corrupt an existing II installation.
5. If an alternate *instance ID* (for example, A2) is specified, we recommend that the *instance PATH* be specified (for example, `/opt/Ingres/IngresA2`).
6. When `ingres_express_install` completes, if the instance will be managed by the "ingres" system user, shut down the instance (`ingstop`), log off from the "root" system user, log on to the "ingres" system user, and execute `.ingXXsh` or `.ingXXcsh` (where XX is the instance ID), which are found in the `$II_SYSTEM/ingres` directory, then start ingres (`ingstart`).
7. `ingres_express_install` will not upgrade an instance. If a previous instance is found, the installation is aborted.

Examples: ingres_express_install Command

This command installs all RPM packages in the current working directory with the default configuration:

```
ingres_express_install
```

This command installs all RPM packages with the default configuration into `/opt/Ingres/IngresII`, but with an instance ID of A1:

```
ingres_express_install A1
```

This command installs all RPM packages with the default configuration into `/opt/Ingres/IngresA2`, with an instance ID of A2:

```
ingres_express_install /opt/Ingres/IngresA2 A2
```

Note: If you set `II_RESPONSE_FILE` to point to a valid response file, then you can install Ingres with the configuration defined in that response file.

Install Ingres Using RPM Commands

As an alternative to using the express install script, you can install Ingres by invoking RPM directly.

To install a single package, invoke RPM with the appropriate installation flags, including the path to the package you want to install, as follows:

```
rpm -ivh path_to_directory/Ingres_package.rpm
```

To install more than one package at a time, pass multiple file names, specifying the path to each package, as follows:

```
rpm -ivh path_to_directory/Ingres_package.rpm  
path_to_directory/Ingres_package2.rpm path_to_directory/Ingres_package3.rpm
```

To install all packages in the same directory, specify the following:

```
rpm -ivh path_to_directory/*.rpm
```

To install a package into a non-default location (that is, with an II_SYSTEM value other than the default), use the `-prefix` flag when invoking RPM. The following command installs the specified package with `II_SYSTEM=/home/ingres/IngresII`:

```
rpm -ivh --prefix=/home/ingres/IngresII path_to_directory/Ingres_package.rpm
```

Note: All packages installed in a single instance must have the same value of II_SYSTEM. If you use the prefix flag to install the base package, you must install all subsequent packages with the same `-prefix` value.

How You Install Ingres with a Custom Configuration at the Command Line

The process for installing Ingres for Linux as a custom configuration at the command line is as follows:

1. Create a response file that defines the configuration.
2. Specify the name of the response file on the `II_RESPONSE_FILE` environment variable.
3. Install the RPM packages by either using the `ingres_express_install` command or invoking RPM directly.

Response File—Define Configuration for Ingres Instance

A response file contains parameters that define how an Ingres instance is to be installed and configured. You can use any name for the response file.

Each entry in the response file must be on a separate line and in the following form:

```
variable=value
```

where *variable* is an Ingres response file parameter name.

Commonly used variables are listed in Installation Parameters (see page 28). A more extensive list is in Response File Parameters (see page 133).

Example: Response File to Install Ingres as a Two-Disk System

The following response file shows Ingres installed on two disks, as follows:

- Disk 1—System files, checkpoint, journal, work, and dump locations, and transaction log
- Disk 2—Databases and backup transaction log

The machine has one CPU, is located in Tokyo, and requires the KANJIEUC character set.

Note: The checkpoint, journal, work, dump, and transaction log locations all reside in the default location for the system files (II_SYSTEM), so you do not need to specify a location.

The response file for this configuration is as follows:

```
II_DATABASE=/disk2  
II_DUAL_LOG=/disk2  
II_TIMEZONE_NAME=JAPAN  
II_CHARSET=KANJIUC
```

Example: Response File to Install Ingres as a Four-Disk System

The following response file shows Ingres installed on four disks, as follows:

- Disk 1—Checkpoint, journal, and dump locations
- Disk 2—System files, transaction log, work files
- Disk 3—Databases
- Disk 4—Backup transaction log

The machine has two CPUs, requires a 500 MB transaction log, and is located in New York. You want the database to comply with the ANSI/ISO Entry SQL-92 standard.

The response file for this configuration is as follows:

```
II_DATABASE=/disk3
II_CHECKPOINT=/disk1
II_JOURNAL=/disk1
II_DUMP=/disk1
II_WORK=/disk2
II_LOG_FILE=/disk2
II_DUAL_LOG=/disk4
II_LOG_FILE_SIZE_MB=500000
II_NUM_OF_PROCESSORS=2
II_TIMEZONE_NAME=NA-EASTERN
II_ENABLE_SQL92=ON
```

Set the II_RESPONSE_FILE Variable

To install a custom configuration, you must set the Ingres environment variable `II_RESPONSE_FILE` to the name of response file that contains the configuration settings. Use the absolute pathname.

To set the `II_RESPONSE_FILE` variable, use the following command:

For bash:

```
export II_RESPONSE_FILE=path_to_directory/response file name
```

For tcsh:

```
setenv II_RESPONSE_FILE path_to_directory/response file name
```

Note: The response file must be in a directory that is globally readable or the install process will fail.

Rebuild RPM Package with Unique Name

To install multiple Ingres instances on one machine, you need a unique set of package names for each instance. You must rebuild each RPM package to include an instance ID that is unique to the machine. You can then install this package using the instructions described in *Install Ingres Using RPM Commands* (see page 36).

To rebuild the package, use the `iirpmrename` command. Multiple RPMs can be specified for renaming.

For example, to rebuild the base package named `Ingres2006-9.0.4.rpm` to use an `II_INSTALLATION` of `XX`, issue the following command:

```
iirpmrename [path_to_directory/]Ingres2006-9.0.4.rpm XX
```

A new RPM package is created in the current working directory called `Ingres-XX-9.0.4.rpm`. The package will install with `II_SYSTEM=/opt/Ingres/IngresXX` and `II_INSTALLATION=XX`.

Note: While you can still override the value for `II_INSTALLATION` using the response file or other methods mentioned above, we do not recommend it.

How You Upgrade Using RPM

Follow these instructions to safely upgrade any Ingres release using RPM.

To upgrade Ingres RPMs

1. Make sure the Ingres instance you want to upgrade is completely shut down and that you have taken a backup, as described in the *Migration Guide*.

Important! Files that you have customized will be lost during the upgrade. Copy your customized files to a safe place. For more information, see the *Migration Guide*.

2. Issue the following command to see the list of Ingres RPM packages installed for the instance to be upgraded:

```
rpm -q --whatrequires core_pkg_name
```

where `core_pkg_name` is the core package name for the release, for example, `ingres2006`.

Note: If you are upgrading an instance that was installed using RPMs that were renamed to embed an instance ID, then use "`core_pkg_name-XX`" instead of the core package name.

Note: All packages must be upgraded at once. Any attempt to perform an upgrade on a single or smaller group of packages will fail.

3. Log in as root and CD to the directory containing the RPM to be used to perform the upgrade. Make sure variable II_CONFIG is not set.
4. Run `rpm -ivh` to install the new RPM packages over the existing instance.
If the instance was originally installed using `--prefix` to change II_SYSTEM from the default, then this parameter must also be used for the upgrade.

For default instance location:

```
rpm -ivh --replacefiles --replacepkgs \  
  complete list of RPM files to be installed
```

For any other instance location:

```
rpm -ivh --replacefiles --replacepkgs --prefix <II_SYSTEM> \  
  complete list of RPM files to be installed
```

5. Remove the RPM package information for the old instance, as follows:

- a. Find the correct packages:

```
rpm -q --whatrequires core_pkg_name
```

- b. If the list returned is satisfactory, remove the packages:

```
rpm -q --whatrequires ca-ingres | xargs rpm -e --justdb
```

- c. Remove the RPM package information for the old core package.
Include the version and the package name, for example:

```
rpm -e --justdb core_pkg_name-x.y.z
```

where x.y.z is the version number (for example 9.0.4).

Notes:

- If you used `rpm -Uvh` with `--prefix` to upgrade an instance, binary and other installed files (for example, `iimerge`) may be removed. Should this occur, simply re-run the same command using the additional flags `--replacefiles` and `--replacepkgs`, and all the missing files will be replaced. Database or configuration files will not be affected.
- For the special case where you are upgrading from a non-NPTL build to a NPTL build with the same version number, you must upgrade using the following command:

```
rpm -ivh --replacefiles --replacepkgs new-package-list
```

Install Ingres for Windows

The installation process on Windows uses an easy to follow setup wizard.

To start the Ingres Setup Wizard

1. Log on to your Windows machine and execute the file setup.exe located in the root directory of the Ingres distribution.

Note: To install Ingres, you must be logged on as a user with administrative privileges.

The first page of the Ingres Setup Wizard is displayed.

2. Respond to the setup wizard dialogs.

If you need more information about a specific item, click the item's information button .

After you provide the required information in the setup wizard dialogs, the installation program installs Ingres.

Separately Installed Components

The Ingres .NET Data Provider and Ingres Documentation are provided as separately installable components. They are displayed as installable components in the setup wizard only if their packages are detected in the installation image you downloaded. When a full Ingres image (Ingres, Ingres .NET Data Provider, and Ingres Documentation) is installed, these three components are listed as separate programs in the Add/Remove Programs feature of the Control Panel in Windows.

Note: If you download the Ingres .NET Data Provider and Ingres Documentation packages to the same directory as the Ingres download, they will appear as installable components.

Advanced Component Selection

The Component Selection page of the setup wizard lets you select a Custom installation.

When Custom is selected, the checkbox "Select advanced customization of components" appears. This option lets you select individual components (rather than default packages) for installation. For example, Vision and network protocols other than TCP/IP are not installed by default, so you can select these individually using the advanced Custom Setup.

This option is recommended for advanced users only.

Perform a Silent Install on Windows

You can use the installer on Windows to create a response file to support an unattended (silent) installation of Ingres.

To create a response file using the Windows installer

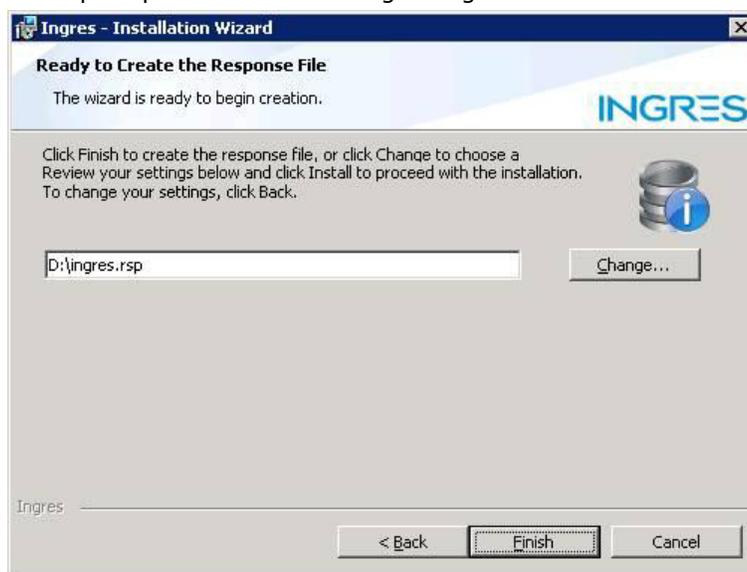
1. From the root directory of the Ingres distribution, execute the following command:

```
setup.exe /c
```

A message informing you that the installer is running in response file mode is displayed.

Click OK to display the first page of the Ingres Setup Wizard.

2. Respond to the Ingres Setup Wizard dialogs to select the platform, components, and configuration settings for your Ingres instance.
3. Review the default storage location and file name for the response file when prompted on the following dialog.

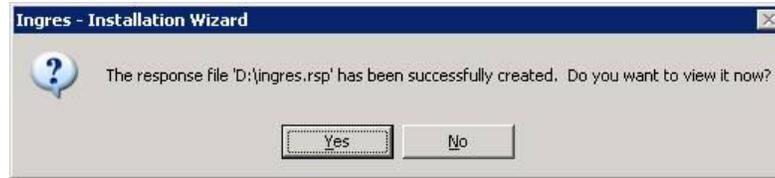


To change the default location or file name, click Change, enter the new information and then click OK.

Note: The response file must be in a directory that is globally readable, or the installation process will fail.

4. Click Finish to create the response file.

A message similar to the following is displayed:



5. Click Yes to review the response file parameter settings in a text editor; otherwise, click No to exit the Ingres Setup Wizard.

A response file can be opened with any text editor and modified manually if changes are required. For details on the parameters, see Response File Parameters (see page 133).

To install Ingres using a response file

Issue the following command from the root directory of the Ingres distribution:

```
setup /r path/response_file_name
```

path

Specifies the full storage location path for the response file.

response_file_name

Specifies the name of the response file.

Example setup /r command

```
C:\Ingres2006\cdimage\setup /r C:\Ingres2006\myresponsefile.rsp
```

Install Ingres for UNIX

Ingres for UNIX can be installed in interactive mode using a form-based utility.

To install Ingres on UNIX using the forms-based Ingres Installation Utility

1. Log in as the root user.
2. Extract all files from the "CD Image" saveset that was downloaded from the Ingres web site to a local directory (or insert the CD-ROM into the drive and mount the drive from a local host, if not mounted automatically).
3. Change to the root directory on the directory where you extracted the "CD Image" saveset files or the CD-ROM, and enter the following command to run the install utility:

```
% install.sh
```

The forms-based Ingres Installation Utility is started.

4. Respond to the install dialogs.
Ingres components are installed on your system.

More information:

Installing Ingres on UNIX (see page 59)

Install Ingres for OpenVMS

To install Ingres on OpenVMS, you execute the VMSINSTAL utility. VMSINSTAL runs the installation and setup scripts, which prompt you for the appropriate directory locations and other necessary information. (For specific information about VMSINSTAL, see your OpenVMS documentation.)

Note: Before running the install program, make a complete backup of your system.

Note: Upon completion, the install utility checks your *current* system resources—that is, the resources available during installation—to ensure that you can run Ingres as configured. You may want to perform the installation while running any non-Ingres programs that will typically run at the same time to determine the resources that will be available to Ingres.

To install the Ingres software on OpenVMS

1. At the operating system prompt, enter

```
$ @sys$update:vmsinstal * distribution_medium
```

The *distribution_medium* is the name of one of the following:

- A physical device, such as a CD-ROM drive, which holds the distribution medium
- A storage directory containing the files optionally transferred from the distribution medium

VMSINSTAL initializes the software environment, defines logical names and symbols, and makes validity checks. It also gives you an opportunity to cancel the install process, if you need to make a backup of your current system. If you choose to continue, all user-defined global and local OpenVMS symbols are deleted at that point.

The program then displays a description and important information about using the installation script.

2. Respond to the program prompts.

Some prompts display default values in brackets. To accept the default, press Return.

Enter the value for II_SYSTEM when you see the following prompt:

Location for this installation:

If the Ingres directory does not exist, the program creates it.

If any installed products require configuration, VMSINSTAL asks:

Do you want to run the configuration IVP following this installation [YES] ?

Enter **y** to run the Setup programs immediately after installation; otherwise, enter **n** to run the Setup programs later.

A final Installation Summary appears.

3. Confirm the installation summary.

VMSINSTAL installs the specified products and verifies that each product was transferred properly from the distribution medium. If you selected products that are dependent on other products, VMSINSTAL automatically installs the necessary products.

The installation finishes or starts executing the setup scripts.

The software is now installed on your system.

More information:

Installing Ingres on OpenVMS (see page 83)

How You Access the Instance on Linux and UNIX

When the installation is complete, the instance is running.

To access your instance, you must source the environment file that was created during installation.

During installation, an environment file is written to the home directory of the operating-system user ID that was defined during installation (the default is "ingres"). The name of the environment file depends on the value of `II_INSTALLATION`.

To source the environment file created during installation, issue the following command.

Note: The following examples assume an operating-system user ID of `ingres`.

For bash:

```
. ~/.ingres/.ingXXbash
```

For tcsh:

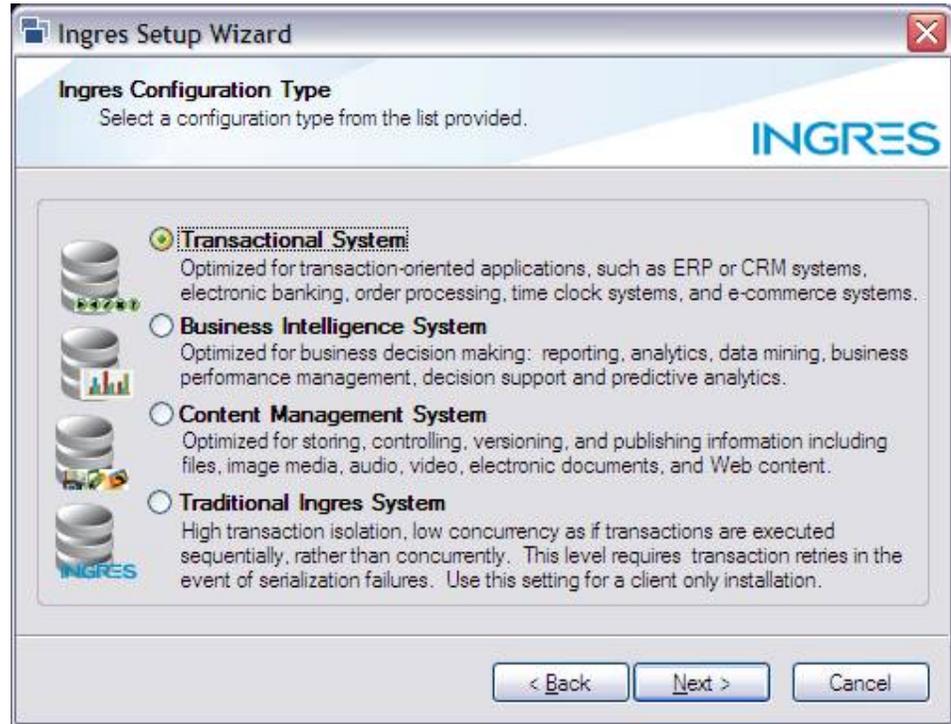
```
source ~/.ingres/.ingXXtsch
```

where `XX` is the instance ID of the instance.

For other users to have access to the instance and the Ingres tools, they must have access to the `.ingXXbash` and `.ingXXtsch` scripts. The scripts can be copied to the home directory of any user.

Configuration Type Parameter Settings

The Ingres Configuration Type page of the installation wizard on Linux and Windows lets you choose a configuration type, as shown in this example from the Windows setup wizard:



Each setting optimizes Ingres configuration parameters based on how Ingres will be used. The selected configuration type is stored as response file parameter `II_CONFIG_TYPE` (see page 144).

Parameter settings for each configuration type are as follows:

Parameter	Transactional	Business Intelligence	Content Management	Traditional
<code>config.date_alias</code>	<code>ansidate</code>	<code>ansidate</code>	<code>ansidate</code>	<code>ingresdate</code>
<code>dbms.*.blob_etab_page_size</code>	16384	16384	16384	2048
<code>dbms.*.cache_dynamic</code>	OFF	ON	OFF	OFF
<code>dbms.*.connect_limit</code>	32	64	256	32
<code>dbms.*.cursor_limit</code>	128	32	128	16
<code>dbms.*.max_tuple_length</code>	0	0	0	2008

Parameter	Transactional	Business Intelligence	Content Management	Traditional
dbms.private.*.p8k.dmf_separate	ON	ON	ON	OFF
dbms.private.*.cache.p16k_status	ON	ON	ON	OFF
dbms.private.*.p16k.dmf_separate	ON	ON	ON	OFF
dbms.private.*.cache.p32k_status (Linux)	ON	ON	ON	OFF
dbms.private.*.cache.p32k_status (Windows)	OFF	OFF	ON	OFF
dbms.private.*.p32k.dmf_separate	ON	ON	ON	OFF
dbms.private.*.dmf_group_size	default	32	default	default
dbms.private.*.p8k.dmf_group_size	default	32	default	default
dbms.*.opf_active_limit	derived	derived	51	derived
dbms.*.opf_joinop_timeout	100	100	0	0
dbms.*.opf_memory	50000000	50000000	23855104	derived
dbms.*.opf_timeout_factor	1	1	10	10
dbms.*.system_isolation	read_committed	read_committed	read_committed	serializable
dbms.*.system_lock_level	row	row	row	default
dbms.*.system_maxlocks	1500	500	default	default
dbms.*.system_readlock	shared	no lock	no lock	shared
dbms.*.table_auto_structure	ON	ON	OFF	OFF
rcp.lock.per_tx_limit (Windows)	4000	15000	15000	750
rcp.lock.per_tx_limit (Linux)	3250	15000	15000	750
rcp.lock.lock_limit	default	325000	325000	default
rcp.log.buffer_count	100	50	200	default

Chapter 3: Starting Ingres

This section contains the following topics:

[Start the Ingres Installation on Linux and UNIX](#) (see page 49)

[Startup and Shutdown on Windows](#) (see page 50)

[Start the Ingres Installation on OpenVMS](#) (see page 53)

[Getting Started with Databases and Applications](#) (see page 53)

Start the Ingres Installation on Linux and UNIX

Use the `ingstart` command to start Ingres.

Note: If you are using a raw device for your transaction log file, you must configure the log file before starting your instance.

Note: If your operating system has shadow passwords, you must install the password validation program (see page 79) before starting a networked DBMS Server instance.

To start the Ingres instance

1. Log on to your system through the system administrator account for the instance.
2. Enter the following command:

```
% ingstart
```

The `ingstart` command checks whether you have sufficient operating system resources to run the Ingres components, and whether the raw log file (if used) is configured. If these conditions are met, `ingstart` starts all servers that are part of your instance.

To stop the Ingres instance

Enter the following command:

```
% ingstop
```

The instance is stopped.

Startup and Shutdown on Windows

You can start up and shut down your Ingres instance on Windows in the following ways:

- Automatically
- Using Ingres Visual Manager
- Using Ingres Service Manager
- Using the Services Window in Windows
- Using the `ingstart` and `ingstop` commands

Note: Due to tightened security on the Microsoft Windows Vista platforms, we must impose the following restriction when using Ingres on Windows Vista: All Ingres tools and applications must be started from the Ingres Command Prompt, which is located in the Ingres Program Group under the Vista Start menu. In addition, to secure the appropriate privileges to execute a server process on Vista, you must always start Ingres as a service.

Start and Stop Ingres with Ingres Visual Manager

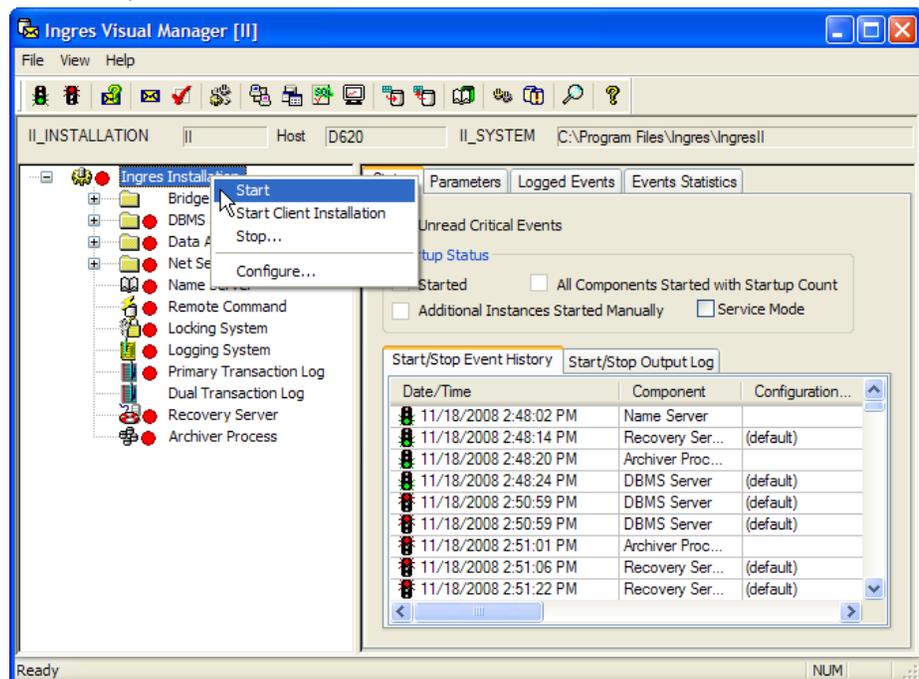
Starting and stopping Ingres is one of the many tasks you can perform with Ingres Visual Manager.

To start and stop Ingres using Ingres Visual Manager

1. Click Start, Programs, Ingres, Ingres Visual Manager.

The Ingres Visual Manager window appears.

2. Right-click the Ingres Installation branch and choose Start, as shown in this example:



Red icons turn green as the various components in the instance are started.

3. Right-click the Ingres installation branch and click Stop.

The instance is stopped.

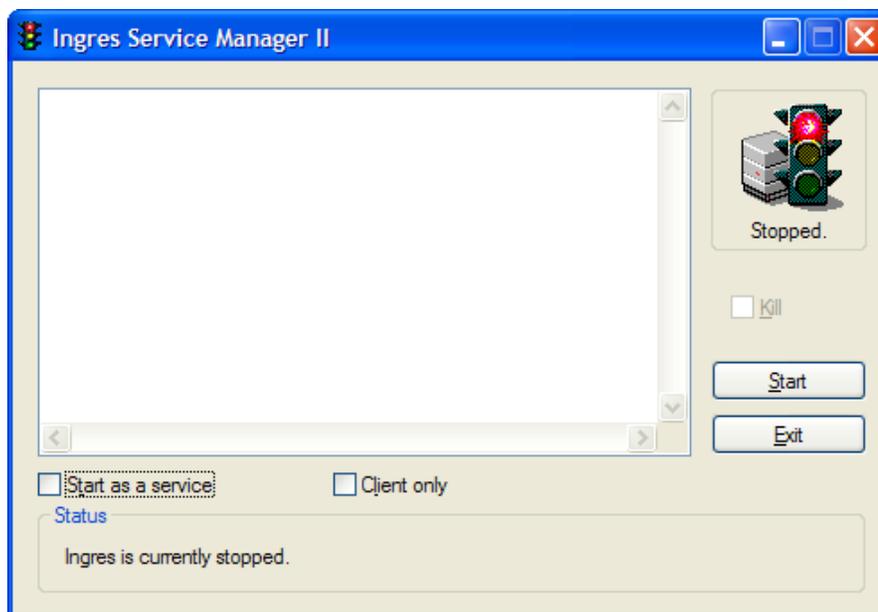
Start and Stop Ingres with the Ingres Service Manager

The Ingres Service Manager lets you start Ingres as a service.

To start and stop Ingres using the Ingres Service Manager

1. Click Start, Programs, Ingres, Ingres Service Manager.

The Ingres Service Manager window appears, as shown here:



2. Select the Start as a Service check box. Select the Client Only check box to start only those components for a client instance. Click Start to start the instance.

The Ingres instance is started, and the output of the startup operation is displayed in the scrollable list box. The Start button changes to Stop.

3. Click Stop

The instance is stopped.

Start the Ingres Installation on OpenVMS

You are ready to start the Ingres installation after you have installed the software and set up the required configuration parameters for each component.

To start your installation

1. Log on to your system through the system administrator account for your installation.

2. Enter the following command:

```
$ @II_SYSTEM: [ingres]ingsysdef.com
```

The symbols needed to start the installation are defined.

3. Enter the following command to start your installation:

```
$ ingstart
```

Ingstart checks that you have sufficient operating system resources to run the Ingres components; if not, it issues an error message and exits. If resources are sufficient, all servers are started.

To stop your installation

Enter the following command:

```
$ ingstop
```

Components of your installation that are running are shut down.

Getting Started with Databases and Applications

Getting started with Ingres includes creating databases and developing database applications. You can connect to Ingres from various application development environments, such as .NET, Java, Eclipse, Python, PHP, and Perl.

For more information on getting started with Ingres, see the *Quick Start Guide*.

Note: Most of the information in the Ingres for Linux *Quick Start Guide* about connecting to Ingres is applicable to the UNIX environment as well.

Chapter 4: Understanding Post-installation Tasks

This section contains the following topics:

[How You Further Customize Your Instance](#) (see page 55)

[How You Prepare Your Installation for General Use](#) (see page 56)

[How You Safely Uninstall Ingres](#) (see page 57)

How You Further Customize Your Instance

After Ingres is installed, configured, and started, you may need to perform the following additional tasks:

- Change default values for some configuration parameters to suit your particular needs or enhance performance.

The default configuration installed by the setup programs is suitable for moderate hardware (about 512 MB of memory) and a moderate number of users (about 32). You may need to change the size of the transaction log file or the number of concurrent users (`connect_limit`).

To change parameter settings, use the Configuration-By-Forms utility (or its equivalent visual tool Configuration Manager, if available) or the `ingsetenv` command.

Note: Some changes, such as to the logging system, should not be made while the DBMS Server is running. Changes made with Configuration-By-Forms while servers are running are not effective until you restart the servers.

- Set up virtual nodes, remote user authorizations, and connection data, as needed, for DBMS Server and client instances. Use the Net Management Utility (`netutil`) or Visual DBA, if available.

If you installed Ingres Net for the first time and did not define the DBMS Server password on the server or client nodes, add authorization entries that enable clients to access databases on the DBMS Server node.

If you are upgrading an existing instance or adding Ingres Net to your instance, your existing definitions remain in effect. For more information, see the *Connectivity Guide*.

- Add client capabilities to a networked DBMS Server by setting up the appropriate Ingres Net client parameters. For details, see the *Connectivity Guide*.
- (UNIX only) Set up and configure a raw log device (see page 75) in place of the default transaction log file.

How You Prepare Your Installation for General Use

After Ingres is installed, you must complete additional tasks so that users can access and use the new installation.

To prepare Ingres for general use, you must complete all or some of these tasks:

- Enable recovery of your master database.

To help recover your master database (iiddb) if it becomes corrupted, make sure that iiddb has been checkpointed and journaling enabled (the default). For details, see the *Database Administrator Guide*.

- Add an automatic startup command.
- Authorize users to start and stop servers.

If you want users other than the Ingres system administrator to be able to start and stop the Ingres servers, you must make changes to the config.dat file. For details, see the *System Administrator Guide*.

- (UNIX and Linux only) Allow access on systems using shadow passwords.
- Allow users to access tools and databases.
- Create databases. For details, see the *Quick Start Guide* or *Database Administrator Guide*.

Note: You do not need to create the iiddb master database because it is created during installation.

- (OpenVMS only) Maintain the installation history log file.

Note: Check the Readme file for any additional tasks necessary for your platform.

More information:

Post-Installation Tasks (UNIX) (see page 75)
Post-Installation Tasks (VMS) (see page 92)

How You Safely Uninstall Ingres

Removing Ingres is an irreversible event with pervasive effects. Any products or applications that share the removed DBMS Server are affected, as follows:

- Any future attempt to connect to this database will fail.
- If you re-install Ingres, you may not be able to reference the data files.

If you know you want to remove an Ingres instance, follow this process:

1. If you want to keep the data files for later use, you must run `unloaddb` against each database before removing Ingres. Doing so will allow you to reference the data files easily if you re-install Ingres later. For details on the `unloaddb` command, see the *Database Administrator Guide* and the *Command Reference Guide*.
2. If you do *not* want to keep the data files, you can run `destroydb` against each database before removing Ingres. The `destroydb` command locates and deletes all data files. For details on the `destroydb` command, see the *Command Reference Guide*.
3. Uninstall Ingres.

Uninstall Ingres for Linux Using the Installation Wizard

To uninstall Ingres for Linux, start the installation wizard and choose the option to modify an existing instance. The option "Remove all components" removes only those files that are distributed with Ingres; it does not remove database and configuration files. To remove all files, select the "Remove all files" option.

uninstall_ingres Command—Uninstall Ingres for Linux

The `uninstall_ingres` command removes any or all instances of Ingres for Linux on a machine. This command invokes RPM. If this command is run without any parameters, it generates a list of standard (non-renamed) installed Ingres RPM packages and prompts for confirmation before removing them.

This command has the following format:

```
uninstall_ingres {[instance ID] | [-a (--all)]} [-y (--yes)] [-c (--clean)]
```

instance ID

Identifies the instance ID embedded in the renamed package names to be removed.

-a

Removes all Ingres RPM packages for *all* Ingres instances. This parameter cannot be used if specifying an instance ID.

-y

Answers yes to all prompts (that is, does not prompt).

-c

Removes the `$II_SYSTEM/ingres` directory after the uninstall operation is complete.

Uninstall Ingres for Windows

The Ingres uninstall program removes all binary files.

To uninstall Ingres

1. Click Start, Settings, Control Panel, Add or Remove Programs.
The Add or Remove programs window appears.
2. Select the Ingres instance or component you want to remove.
Note: If multiple instances installed, be sure to select the one you want to remove. Each instance is identified by a unique instance name.
Note: Ingres .NET Data Provider and Ingres documentation, if installed, are separately installed products and must be individually selected for removal.
3. Click Add/Remove.
The Ingres installation wizard is started.
4. Follow the installation wizard instructions to remove Ingres.
Ingres binary files and any data files residing in `II_SYSTEM` are removed.

Appendix A: Installing Ingres on UNIX

This section contains the following topics:

[How You Prepare for Installation on UNIX](#) (see page 59)

[How You Install and Configure Ingres in a UNIX Environment](#) (see page 62)

[Set TERM_INGRES](#) (see page 63)

[Installation Modes](#) (see page 63)

[Extract Files from the Distribution](#) (see page 64)

[ingres_express_install Command—Install Ingres Quickly](#) (see page 64)

[Install Ingres in Interactive Mode](#) (see page 65)

[Install Ingres in Command Line Mode](#) (see page 65)

[Run the Setup Programs](#) (see page 70)

[How You Install and Configure NFS Clients](#) (see page 72)

[Post-Installation Tasks](#) (see page 75)

How You Prepare for Installation on UNIX

You should thoroughly understand the issues regarding system resources, storage locations, and other configuration parameters before installing Ingres. If you are not sure, have someone more knowledgeable in these areas perform the installation planning.

Before installing Ingres on UNIX, do the following:

- Verify system requirements. (For requirements, see the Readme.)
- Know how you want to configure your instance.
- Create the system administrator account.
- Create the directory for system files for Ingres and Ingres tools.
- Decide file locations.

How You Create the System Administrator Account on UNIX

Only the system administrator (instance owner) can install Ingres, therefore, you must create this account before installing the product.

Follow this process to create the system administrator account:

1. Log in to your system through the root account.

If you do not have access to this account, have the UNIX system administrator complete this procedure or log in for you.

2. Set up an account with a login name of your choice. (The default is "ingres.")

This account should have a user identification code (UID) greater than 100.

3. Add this user to the kmem group in the /etc/group file, as described in your operating system documentation.
4. Make sure that /dev/kmem is in the kmem group and that its permissions are set to "group read." For example:

```
%ls -lgL /dev/kmem  
crw-r----- 1 sys 13, 1 Jun 19 09:05 /dev/kmem
```

Note: Check the Readme file for your operating system for any variation in this procedure.

Major Configuration Options

Initially, you can configure your installation as one of the following major types. Later, you can install other products or modify parameter settings to add more capabilities to your installation. For example, you can add networking capabilities to a stand-alone installation that is connected to a network. Or you can add client capabilities to an installation that you initially configured as a networked DBMS Server.

The major configuration options are as follows:

Stand-alone DBMS Server

Provides only local access to local databases. This instance includes the Name Server, DBMS Server, and its own set of tools.

Required products: DBMS Server, tools

Networked DBMS Server

Allows remote clients to access its databases through a network. (If tools are installed, local users also can access its databases.)

This instance includes the Name Server, DBMS Server, Communications Server, Data Access Server, ODBC driver, and JDBC driver.

By installing Ingres tools and then modifying the Ingres Net connection data, you can add outgoing client capabilities to a networked DBMS Server instance. Doing so enables it to act both as a client of a remote DBMS Server and as a DBMS Server to its own remote clients.

Required products: DBMS Server, Ingres Net, ODBC driver, JDBC driver, Ingres tools (optional, but required for local access to local databases)

Networked DBMS Server with Ingres Star

Allows access to multiple databases—local and remote, Ingres and non-Ingres—simultaneously. This instance includes Ingres Star, Name Server, DBMS Server, Communications Server, Data Access Server, ODBC driver, and JDBC driver. It may also contain Ingres tools.

Required products: DBMS Server, Ingres Net, ODBC driver, JDBC driver, Ingres Star, Ingres tools (optional)

Client

Has its own set of Ingres tools and accesses databases on a networked DBMS Server instance on a remote node. This installation includes the Name Server, Communications Server, Data Access Server, ODBC driver, JDBC driver, and Ingres tools. It does not contain the DBMS Server.

You can configure a networked DBMS Server instance as a client of another DBMS Server on a remote node. To do so, install Ingres tools on the DBMS Server, and then add client capabilities by modifying the Ingres Net connection data.

You can set up a client in a different environment to access an Ingres database in your current environment. For example, a client in a Windows environment can access a database in a Linux environment. For more information, see the *Connectivity Guide*.

Required products: Ingres Net, ODBC driver, JDBC driver, Ingres tools

Network File System (NFS) Client

Uses a Network File System to share all Ingres files on a networked DBMS Server instance. This instance contains no Ingres files on its own node.

Required products: A networked DBMS Server instance on a node in the Network File System. The `ingmknfs` utility must be run to create the NFS client instances and configure the DBMS Server for NFS-client use.

How You Install and Configure Ingres in a UNIX Environment

The process for installing and configuring Ingres on UNIX is as follows:

1. **Set environment variable**—Define your terminal type, as described in Set `TERM_INGRES` (see page 63).
2. **Install the software**—Install the software in the correct locations with the correct permissions.
3. **Run setup programs**—Set required configuration parameters for installed components by running the setup programs.
Note: This step is necessary only if the install procedure did not include this step, for example, if you used the Install method in Interactive mode.
4. **Start Ingres**—Start Ingres on your system so the system administrator can access it.
5. **Customize your instance**—Set optional configuration parameters to allow Ingres to run as desired.
6. **(Optional) Set up raw log file**—Set up a raw log file to be used in place of the default transaction log file, as described in Transaction Log File as a Raw Device (see page 75).
7. **Prepare Ingres for general use**—Perform additional tasks needed to prepare the product for its users, such as creating an automatic boot command, authorizing users, and creating databases.

Set TERM_INGRES

The TERM_INGRES environment variable defines the terminal type you are using. You can enter a value for the TERM_INGRES environment variable in response to a prompt during installation or prior to running the install program.

To set TERM_INGRES

Enter the following command at the operating system prompt:

```
% setenv TERM_INGRES Ingres_terminaltype
```

For example, the following command defines your terminal as a VT100.

For C shell:

```
% setenv TERM_INGRES vt100
```

For Bourne shell:

```
$ TERM_INGRES=vt100; export TERM_INGRES
```

Installation Modes

Ingres on UNIX can be installed using one of the following modes:

- **Express Install mode**—Installs Ingres in the default location using default configuration values, without any prompts.
- **Interactive mode**—Installs Ingres through a forms-based utility that lets you make decisions throughout the process. Your actual installation and setup script output is saved to a file called install.log in the \$II_SYSTEM/ingres files directory.
- **Command line mode**—Installs Ingres through a command (and any optional flags) that you enter at the operating system prompt. The installation procedure runs without interruption.

Extract Files from the Distribution

The Ingres distribution is typically a “CD Image” saveset in tar or tgz format.

To extract the files from the distribution

1. Extract the `ingres.tar` file from the saveset.
2. Set environment variable `II_DISTRIBUTION` to point to `ingres.tar`. For example, in Bourne or Bash shells:

```
$ II_DISTRIBUTION pathname; export II_DISTRIBUTION
```

where *pathname* is the path to the `ingres.tar` file.
3. From `$II_SYSTEM/ingres` directory, extract the install files with the following command:

```
tar xvf $II_DISTRIBUTION install
```
4. Change to `$II_SYSTEM/ingres/install` directory, and issue the appropriate command to install Ingres.

ingres_express_install Command—Install Ingres Quickly

The `ingres_express_install` command quickly installs all Ingres packages.

If you set `II_RESPONSE_FILE` to point to a valid response file, then you can install Ingres with the configuration defined in that response file.

This command has the following format:

```
ingres_express_install [instance_ID]
```

instance ID

Specifies the Ingres instance ID to use for the instance.

Examples: ingres_express_install Command

This command installs all packages in the current working directory with the default configuration:

```
ingres_express_install
```

This command installs all packages with the default configuration, but with an instance ID of `A1`:

```
ingres_express_install A1
```

Install Ingres in Interactive Mode

In interactive mode, you install Ingres using a form-based utility.

To install Ingres on UNIX using the forms-based Ingres Installation Utility

1. Log in as the root user.
2. Extract all files from the "CD Image" saveset that was downloaded from the Ingres web site to a local directory (or insert the CD-ROM into the drive and mount the drive from a local host, if not mounted automatically).
3. Change to the root directory on the directory where you extracted the "CD Image" saveset files or the CD-ROM, and enter the following command to run the install utility:

```
% install.sh
```

The forms-based Ingres Installation Utility is started.

4. Respond to the install dialogs.
Ingres components are installed on your system.

Install Ingres in Command Line Mode

In command line mode, you can completely automate and customize your install by supplying the appropriate arguments to the `ingbuild` command at the operating system prompt.

Set II_DISTRIBUTION

The II_DISTRIBUTION environment variable stores the full path name for the device or file containing the Ingres distribution.

You can enter a value for the II_DISTRIBUTION environment variable in response to a prompt during installation or prior to running the install program.

Note: For faster file transfer during the installation process, you can copy the contents of the distribution medium to a disk file as an intermediary step, and then specify the disk file as the *pathname* for II_DISTRIBUTION.

To set II_DISTRIBUTION

Enter the following command at the operating system prompt:

For C shell:

```
% setenv II_DISTRIBUTION pathname
```

For Bourne shell:

```
$ II_DISTRIBUTION=pathname; export II_DISTRIBUTION
```

Ingbuild Command—Install Ingres

The `ingbuild` command installs Ingres.

`ingbuild` runs in interactive mode if you specify no command line arguments or only `-mkresponse` or `-exresponse`.

`ingbuild` runs in batch mode (without interruption) if you specify a distribution medium. It also automatically starts any needed setup programs for the installed products, unless the `-nosetup` option is specified.

This command has the following format:

```
ingbuild [flags] [distribution_medium]
```

flags

Specifies zero or more install flags, as follows:

-all

Installs all products (default). This flag is valid when a distribution medium is specified.

-install=*list*

Installs only those products you list. Use a comma to separate each item in the list. This flag is valid when a distribution medium is specified.

-products

Displays a list of the products on the distribution medium, which you can then specify with the `-install` flag or `-version` flag. This flag should not be used in combination with other flags.

-version[=*list*]

Displays the version number of the instance (when specified without *list*) or displays the version of the requested packages (when specified with *list*). Use commas to separate items in the list. This flag should not be used in combination with other flags.

-nosetup

Does not run setup programs. This flag is valid when a distribution medium is specified.

-express

Runs setup programs using default values. This flag is valid when a distribution medium is specified.

-acceptlicense

Accepts Ingres license and bypasses license prompt. This flag is valid when a distribution medium is specified.

-ignore

Ignores any errors encountered during file copy. This flag is valid when a distribution medium is specified.

-help

Displays ingbuild command options. This flag is valid when a distribution medium is specified.

-mkresponse

Creates a response file that contains install parameters, but does not install the product. This flag is valid only in an interactive install, and is mutually exclusive with the -exresponse flag.

-exresponse

Reads installation parameters from a response file during an interactive install. This flag is valid only in an interactive install and is mutually exclusive with the -mkresponse flag.

-file=*filename*

Identifies a response file name other than the default for use with -mkresponse or -exresponse.

Default: ingrsp.rsp

distribution_medium

Specifies the name of the distribution medium on which the software to be installed resides. A batch, rather than interactive, install is executed. This parameter is required with the -all, -install, -express, -ignore and -nosetup flags. If a distribution medium is provided with no flag, -all is assumed.

The distribution medium is one of the following:

- The ingres.tar file on the CD-ROM or from the “CD-ROM image” saveset, with full path (for example: /cdrom/ingres.tar)
- A physical device, such as a magnetic tape drive
- A disk file containing an ingres.tar image copied from the distribution medium

Examples: ingbuild command

- This command displays ingbuild options:
% ingbuild -help
- This command lists all product names before installing them:
% ingbuild -products
- This command installs all products:
% ingbuild -all *distribution_medium*

Run the Setup Programs

The setup programs configure the installed components.

You must run the setup programs if the installation process did not perform this step if you used the Install method in Interactive mode—or if you want to change the default values.

After installing Ingres components, you can run the setup programs at any time from the Currently Installed Components screen.

The order in which you run the setup programs is important. For a networked DBMS Server instance, run the setup program for the DBMS Server **before** running Ingres Net Setup. Ingres Net can then use several of the values you specified for the DBMS Server. Run Ingres Star setup last.

To run a setup program

1. Enter the appropriate command, as follows:

```
$II_SYSTEM/ingres/utility/program_name
```

where *program_name* is the name of the setup program, as follows:

iisudbms

DBMS Server setup

iisuc2

C2 Security auditing

iisunet

Ingres Net setup

iisudas

Data Access Server setup

iisubr

Ingres Bridge setup

iisuabf

Applications-By-Forms setup

iisurep

Ingres Replicator setup

iisuodbc

ODBC Driver setup

iisustar

Ingres Star setup

iisurms

RMS Gateway setup

iisukerberos

Kerberos setup

iisusupp32

32-bit support for 64-bit Ingres setup. Valid on reverse hybrid platforms only.

The Setup program for each component displays a description of its purpose and any other important information you should know before setting up the software.

2. Respond to the program prompts.

To accept a default value (which sometimes appears in brackets after the prompt), press Return; otherwise, enter an appropriate value and press Return.

If you do not upgrade all databases during setup, you can upgrade specific databases later, using the `upgradedb` command (which also runs the utility that upgrades the tools catalogs). For more information, see the *Migration Guide*.

When the Setup programs complete, you are returned to the operating system prompt. Your Ingres installation is now configured.

How You Install and Configure NFS Clients

You create and configure NFS client instances using the `ingmknfs` utility, rather than the install and setup programs.

The process for installing and configuring an NFS client is as follows:

1. Install the Ingres DBMS Server, Ingres Net, and Ingres tools components as a networked DBMS Server.
2. Run the `ingmknfs` utility to create the NFS client instances and configure the DBMS Server for NFS-client use (that is, create the NFS admin directory and related NFS-client files).

You can issue the `ingmknfs` command in either of the following ways:

- On each NFS client, individually. With this method, you can establish Ingres Net authorizations by responding to program prompts.
- On the Ingres DBMS Server installation only, using a parameter list of all the NFS client instances you want to set up. With this method, you must run the `netutil` utility on each NFS client to establish Ingres Net authorizations. You must establish an instance owner account that has the same numeric ID (UID) as the instance owner account on the DBMS Server.

Set Up Each NFS Client Individually

To set up each NFS client individually, follow these steps:

1. Make sure you have run the Ingres DBMS Server and Ingres Net setup programs on the Ingres DBMS Server instance, as described in [Run the Setup Programs](#).
2. Define `II_SYSTEM` as it appears on the client, where *pathname* is the path for the directory in the client's file system that is the NFS-mounted, shared `II_SYSTEM` directory on the Ingres DBMS Server instance.

For C shell:

```
% setenv II_SYSTEM pathname
```

For Bourne shell:

```
$ II_SYSTEM=pathname; export II_SYSTEM
```

Details are described in [NFS Client Parameters \(UNIX\)](#).

3. Set your path appropriately.

For C shell:

```
% set path=($II_SYSTEM/ingres/{bin,utility} $path)
% rehash
```

For Bourne shell:

```
$ PATH=$II_SYSTEM/Ingres/bin:$II_SYSTEM/ingres/ utility:$PATH
$ export PATH
```

4. Issue the following command on each NFS client at the operating system prompt:

```
% ingmknfs
```

5. Provide the following information in response to program prompts.

- Installation code for this client that is unique on this node
- World region in which this client is located
- Name of the time zone in which this client is located

The program then asks:

Do you want to authorize any users at this time? (y/n) [y]

6. Enter **y** to create an authorization entry at this time for yourself or for all users, to enable access to the default Ingres DBMS Server. When prompted to do so, enter the installation password for this client's Ingres DBMS Server (host) instance.

Enter **n** if you prefer to create the authorization entry later, using either the `netutil` utility (see the *Command Reference Guide*), or the Network Utility or Visual DBA (see the *Connectivity Guide*).

When done, you are returned to the operating system prompt.

7. Start up the Ingres DBMS Server instance and each NFS client instance with Ingres Visual Manager or the `ingstart` command on each node, as described in Start the Ingres Installation.
8. Make sure that the Ingres startup procedure has succeeded and the Name Server and Communications Server are now running on each client node.

Set Up NFS Clients from the DBMS Server

To set up one or more NFS clients from the server, follow these steps:

1. Make sure you have run the setup program for the Ingres DBMS Server and Ingres Net components on the DBMS Server instance, as described in Run the Setup Programs.

2. From the DBMS Server node, enter the following command at the operating system prompt, in which *host* is the name of each NFS client instance that you want to set up:

```
% ingmknfs host {host}
```

For example, if your DBMS Server node is "sales" and your client nodes are "region1," "region2," and "region3," you would enter the following command on the DBMS Server node:

```
% ingmknfs region1 region2 region3
```

3. Provide the following information in response to the program prompts. Refer to your completed NFS Client Worksheet.
 - Value of II_SYSTEM as it appears on the client, as described in NFS Client Parameters.
 - Installation code for this client that is unique on this node
 - World region in which this client is located
 - Name of the time zone in which this client is located

When done, you are returned to the operating system prompt.

4. Define II_SYSTEM on the client, where *pathname* is the path for the directory in the client's file system that is the NFS-mounted, shared II_SYSTEM directory on the DBMS Server instance:

```
% setenv II_SYSTEM pathname
```

5. Set the path on the client.

For C shell:

```
% set path=($II_SYSTEM/ingres/{bin,utility}) $path  
% rehash
```

For Bourne shell:

```
$ PATH=$II_SYSTEM/ingres/bin:$II_SYSTEM/ingres/utility:$PATH  
$ export PATH
```

6. Start up the DBMS Server instance and each NFS client instance with Visual Manager or the `ingstart` command on each node, as described in Start the Ingres Installation.
7. Make sure that the Ingres startup procedure has succeeded and the Name Server and Communications Server are now running on each NFS client node.

8. On each NFS client node, establish Ingres Net authorizations using either the `netutil` utility (see the *Command Reference Guide*), or the Network Utility or Visual DBA (see the *Connectivity Guide*).

Post-Installation Tasks

Post-installation tasks include the following:

- Create and configure the transaction log file as a raw device
- Add an automatic startup command
- Allow access on systems using shadow passwords
- Establish user access to tools and databases

Transaction Log File as a Raw Device

The transaction log file can be a raw device rather than an ordinary file system file. A raw device is a disk partition without a file system. This avoids file system overhead for disk input and output. Ingres supports use of a raw device for the transaction log files as a high-performance option.

By default the install process creates the log file as an ordinary file. After Ingres is installed, you can create and configure the transaction log file as a raw device.

Location of Your Raw Log File (UNIX)

The following restrictions apply to the locations for a raw log file:

- They must be in their own partitions.
- They must not contain a file system.
- They must not be raw devices used by another program.

Create and Configure the Raw Log File (UNIX)

After Ingres is installed, you can create and configure the transaction log file as a raw device.

Note: Check the Readme file for any variations to this procedure that are specific to your platform.

To create and configure a raw log file

1. Log in to your system as the root account.

(If you do not have access to this account, have the UNIX system administrator complete this procedure or log in for you.)

2. Create the raw devices in the locations you have chosen.

For more information about creating a raw device, see the UNIX documentation.

3. Use the ls command to verify that you have two special devices available for each raw location. Enter the following command:

```
ls -l /dev
```

This command should return output similar to this example:

```
crw-r----- 1 root 17, 3 Dec 16 12:52
/dev/rsd1d
brw-r----- 1 root 11, 3 Dec 16 12:52
/dev/sd1d
```

4. Go to the location of your instance's system files.

For C shell:

```
% setenv II_SYSTEM dirname
% set path = ($II_SYSTEM/ingres/{bin,utility} $path)
% cd $II_SYSTEM/ingres
```

For Bourne shell:

```
$ II_SYSTEM=dirname
$ export II_SYSTEM
$ PATH=$II_SYSTEM/ingres/utility:$II_SYSTEM/ingres/
  bin:$PATH
$ export PATH
$ cd $II_SYSTEM/ingres
```

5. Issue the following command:

```
% mkrawlog
```

The mkrawlog command:

- Requests the full path name of the raw device files.
- Sets Ingres as the owner of all devices.
- Checks that the locations and sizes of the raw devices are valid.
Note: All raw partitions of a multi-partition raw log file must be the same size.
- Sets up file system links to the raw log files in the log subdirectory.

The raw log file is configured.

Automatic Startup Command

You can have Ingres start up and shut down automatically. To add an automatic startup command, you must know the startup mechanism your environment uses. See the administration manual for your UNIX environment, or ask your system administrator to assist.

Typically, UNIX startup scripts are organized either in directories by run-level, or in startups scripts in /etc.

Note: For additions or variations necessary for your platform, see the Readme file.

If your environment uses the "directory-by-run-level" method, we recommend that you sequence your Ingres script so that it starts late in the startup sequence, and shuts down early in the shutdown sequence.

If your environment uses one or a few startup scripts, edit the rc.local script. Look for a shell function called localrc, or simply add the Ingres startup near the end.

When writing the Ingres startup script, make sure that you set all necessary shell environment variables, and run the ingstart command as instance owner. At a minimum, you must set II_SYSTEM and PATH. In most UNIX environments, you must also set the shared library path (LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIBPATH, SHLIB_PATH, or SH_LIB_PATH).

For example:

```
II_SYSTEM=/inghome; export II_SYSTEM

PATH=$PATH:$II_SYSTEM/ingres/bin:$II_SYSTEM/ingres/utility; export PATH

LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$II_SYSTEM/ingres/lib
export LD_LIBRARY_PATH

su ingres -c "ingstart"
```

If shared libraries are not supported on your operating system, there is no need to set the shared library path variable.

Depending on your UNIX environment, you may want to redirect the output of the ingstart command to a file or to /dev/console.

You must add the appropriate commands to the system startup file on each machine running a DBMS Server or client.

Allow Access on Systems Using Shadow Passwords

If your operating system has shadow passwords and you are not using the Ingres installation password authorization, you must install the password validation program to allow access to the networked DBMS Server.

To install the password validation program

1. Log in as root.
2. Run `$II_SYSTEM/ingres/bin/mkvalidpw`.

The `mkvalidpw` script installs the password authorization program used by the DBMS Server.

How You Establish User Access to Tools and Databases

The install process identifies only the owner of the system administrator account to your instance to permit authorized access to the databases.

To enable users to access tools and databases, the system administrator must do the following:

- Edit the user login files to facilitate user access to the tools needed to query the databases.
- Use the `accessdb` command or `CREATE USER` statement to authorize users to access specific databases. For more information, see the *Security Guide*.

Edit User Login File to Facilitate User Access to Tools

To facilitate user access to Ingres tools when they log in to the system, add the following access commands to the users' login files, where *installation_location* is the storage location for your instance's system files (executables).

For C shell, add the following command to each user's `“.cshrc”` file:

```
% setenv II_SYSTEM installation_location
% set path=($II_SYSTEM/ingres/bin $path)
```

For Bourne shell, add the following command to each user's `“.profile”` file:

```
$ II_SYSTEM=installation_location; export II_SYSTEM
$ PATH=$II_SYSTEM/ingres/bin:$PATH; export PATH
```

Edit User Login File to Set Up Shared Library Paths

To support shared libraries on your operating system, include the following command in each user's path. For the name of the library variable (*lib_variable*) for your platform—for example, LD_LIBRARY_PATH, LD_LIBRARY_PATH_64, LIBPATH, SHLIB_PATH, or SH_LIB_PATH—see the Readme.

For C shell:

```
% setenv lib_variable  
/lib:$II_SYSTEM/ingres/lib:$lib_variable
```

For Bourne shell:

```
$ lib_variable=/lib:$II_SYSTEM/ingres/lib: $lib_variable  
$ export lib_variable
```

View Contents of Your Current Installation

After you have completed the install process and started the instance, you can use the install utility with your current instance.

To view the contents of your current instance

1. Log in to your system through the instance owner account.
2. Change your active directory to II_SYSTEM/ingres:

```
% cd $II_SYSTEM/ingres
```

3. Enter the following command at the operating system prompt:

```
install/ingbuild
```

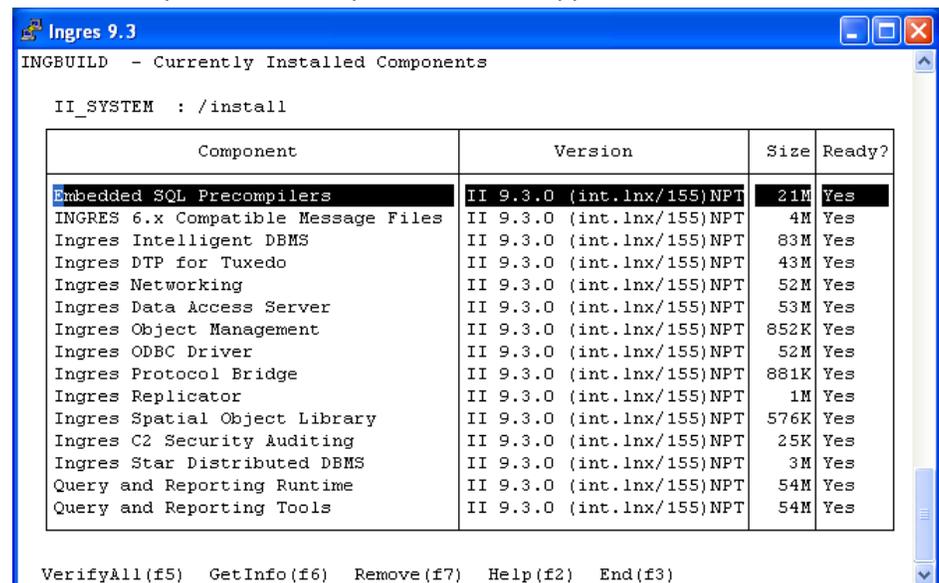
The initial screen of the install utility appears, with the following menu options:

```
CustomInstall PackageInstall Current Help Quit
```

Note: The Current option appears only if products are installed.

4. Choose Current.

The Currently Installed Components screen appears.



All Ingres components that are installed are listed.

Currently Installed Components Screen

The Currently Installed Components screen of the Install utility lets you perform various functions on your current installation, including the following:

- View the contents of your installation
- Verify the components you installed
- Add or remove components from your installation
- Run the setup program for components that were not set up during installation

Appendix B: Installing Ingres on OpenVMS

This section contains the following topics:

- [How You Install and Configure Ingres \(First Time Installers\)](#) (see page 83)
- [How You Install and Configure Ingres \(Experienced Installers\)](#) (see page 84)
- [Define the TERM_INGRES Logical](#) (see page 85)
- [Major Configuration Options](#) (see page 86)
- [Multiple Instances on One Node](#) (see page 87)
- [File Locations and Logicals](#) (see page 88)
- [Owner of the System Administrator Account](#) (see page 90)
- [Run the Setup Programs](#) (see page 91)
- [Post-Installation Tasks](#) (see page 92)
- [Ingres Cluster Solution for OpenVMS](#) (see page 96)

How You Install and Configure Ingres (First Time Installers)

The process for installing and configuring Ingres for OpenVMS for the first time on a node is as follows:

1. (Optional) Define a default value for your terminal type prior to running the install program by defining the TERM_INGRES logical (see page 85).
2. Install the software as described in Install Ingres for OpenVMS (see page 44).

This step copies the contents of the distribution onto your system in the correct locations with the correct permissions. If using the Express Install, it also automatically sets required configuration parameters for installed components to default values.

3. Run the Installation Verification Procedure (IVP) setup programs (see page 91) (Standard Install only).

This step sets required configuration parameters for installed components.

Note: On a cluster installation, the setup programs must be run on every node. However, after completing the install and setup phase on just one node, you can start Ingres to test your installation. When you are satisfied with the results, you should shut down the installation and complete the setup programs on the other nodes.

4. Start the Ingres Installation on OpenVMS (see page 53).
This step starts Ingres on your system so the system administrator can access it.
5. Customize your installation.
This step sets optional configuration parameters to allow Ingres to run as desired.
6. Prepare Ingres for general use.
In this step, you perform additional tasks needed to prepare Ingres for its users, such as creating an automatic boot command, authorizing users, and creating databases.

How You Install and Configure Ingres (Experienced Installers)

If you are experienced at installing Ingres for OpenVMS, you may want to follow this abbreviated process, which is a summary of the major steps.

The major steps to install Ingres on OpenVMS are as follows:

1. Define any logicals (see page 88) for Ingres locations.
2. Ensure you have adequate system resources.
3. Shut down your existing installation, if running, with the `ingstop` utility.
4. Enter the following command from a fully privileged account to start the VMSINSTAL utility:

```
$ @sys$update:vmsinstal * distribution_medium
```

The *distribution_medium* is described in Install Ingres for OpenVMS (see page 44).

5. Respond to prompts on your screen to install the software, and to run IVP Setup programs, if necessary.

Note: On a cluster installation, the VMSINSTAL utility should be run on only one node, but the setup programs must be run on every node.

6. Start your installation by entering:

```
$ ingstart
```
7. Customize your installation further, if appropriate.
8. Prepare your installation for other users.

Define the TERM_INGRES Logical

The Ingres logical TERM_INGRES defines the type of terminal you are using. You can enter a value for TERM_INGRES in response to prompts during the install process, or optionally, prior to running the install program. For information on supported terminal types, see the *Database Administrator Guide*.

To set the TERM_INGRES logical before installation

Enter the following command at the operating system prompt:

```
$ define TERM_INGRES ingres_terminaltype
```

For example, the following command defines your terminal as a VT100f:

```
$ define TERM_INGRES vt100f
```

Major Configuration Options

Initially, you can configure your installation as one of the following major types. Later, you can install other products or modify parameter settings to add more capabilities to your installation. For example, you can add networking capabilities to a stand-alone installation that is connected to a network. Or you can add client capabilities to an installation that you initially configured as a networked DBMS Server.

The major configuration options are as follows:

Stand-alone DBMS Server

Provides only local access to local databases. This instance includes the Name Server, DBMS Server, and its own set of tools.

Required products: DBMS Server, tools

Cluster

Runs on one or more nodes in a cluster environment, which provides access to local databases from any of the nodes running Ingres.

This installation includes the Name Server, DBMS Server, and its own set of Ingres tools. It does not require a Communications Server.

Local databases, transaction log files, and other Ingres areas must be located on commonly accessible cluster devices.

If an Ingres node in the cluster installation crashes, any incomplete transactions on the failed node are recovered by Ingres processes.

Required products: DBMS Server, Ingres tools

Networked DBMS Server

Allows remote clients to access its databases through a network. (If tools are installed, local users also can access its databases.)

This instance includes the Name Server, DBMS Server, Communications Server, Data Access Server, ODBC driver, and JDBC driver.

By installing Ingres tools and then modifying the Ingres Net connection data, you can add outgoing client capabilities to a networked DBMS Server instance. Doing so enables it to act both as a client of a remote DBMS Server and as a DBMS Server to its own remote clients.

Required products: DBMS Server, Ingres Net, ODBC driver, JDBC driver, Ingres tools (optional, but required for local access to local databases)

Networked DBMS Server with Ingres Star

Allows access to multiple databases—local and remote, Ingres and non-Ingres—simultaneously. This instance includes Ingres Star, Name Server, DBMS Server, Communications Server, Data Access Server, ODBC driver, and JDBC driver. It may also contain Ingres tools.

Required products: DBMS Server, Ingres Net, ODBC driver, JDBC driver, Ingres Star, Ingres tools (optional)

Client

Has its own set of Ingres tools and accesses databases on a networked DBMS Server instance on a remote node. This installation includes the Name Server, Communications Server, Data Access Server, ODBC driver, JDBC driver, and Ingres tools. It does not contain the DBMS Server.

You can configure a networked DBMS Server instance as a client of another DBMS Server on a remote node. To do so, install Ingres tools on the DBMS Server, and then add client capabilities by modifying the Ingres Net connection data.

You can set up a client in a different environment to access an Ingres database in your current environment. For example, a client in a Windows environment can access a database in a Linux environment. For more information, see the *Connectivity Guide*.

Required products: Ingres Net, ODBC driver, JDBC driver, Ingres tools

Multiple Instances on One Node

You can have more than one installation on the same stand-alone computer or network node. Each installation on that computer or node must have a unique installation code.

A single node can contain one system-level DBMS Server or client installation and many group-level installations. The system-level installation is the “main” installation, which may be accessed by all authorized users. Each group-level installation will be the default for the members of a given group with a particular user identification code (UIC). Non-group members may use such an installation by running a script that is provided for that purpose.

Important! Assigning multiple instances to the same group UIC corrupts data and the logging and locking system.

System-level and group-level installations use different logical name tables to store Ingres location logicals (for example, II_DATABASE). In a system-level installation, these logicals are set at the system level and are stored in the LNM\$SYSTEM table. In a group-level installation, logicals are set at the group level and are stored in the appropriate LNM\$GROUP table. For instructions on how individual users can switch between system-level and group-level installations, see the *System Administrator Guide*.

The Ingres system administrator must maintain and keep separate any multiple instances on the same node. To help you keep track of these instances, the “ii_installs.com” script and the installation history log file “ingres_installations.dat,” are provided in the distribution.

File Locations and Logicals

You must define logicals to point to a disk and optional directory location where the various files will be stored.

When specifying file locations other than II_SYSTEM, you must use a defined *concealed logical name* rather than the actual disk or directory specification. A *concealed logical name* is a name that you define to represent a particular directory specification. The install program stores the concealed logical name in the appropriate Ingres logical (for example, in II_DATABASE). It then uses your definition of the concealed logical to ascertain the actual location in which it will create the appropriate directory tree. This makes it easier to change locations later, if necessary. For details, see the *System Administrator Guide*.

The II_SYSTEM logical should be defined as either a concealed logical or as a device, if that device is a concealed logical. There is no advantage to using a concealed logical for II_SYSTEM, because the install program stores only the translated real device and optional directory specification in the II_SYSTEM logical. For more information, see the appendix "General Reference."

You must define any logicals other than II_SYSTEM that you use as both "rooted" and "concealed" at the system or group level as appropriate for your installation. A "rooted" logical can be used directly as if it were a device name. A "concealed" logical is one that is not translated any further; that is, the definition is concealed from programs that use them.

Define II_SYSTEM as a Logical

To define II_SYSTEM as a logical, enter one of the following commands at the operating system prompt, as appropriate:

- Group-level installation (while logged into an account in the appropriate group and the GRPNAM privileges):

```
$ define/group/exec/translation=concealed -  
  II_SYSTEM disk:[dir.]
```

- System-level installation:

```
$ define/system/exec/translation=concealed -  
  II_SYSTEM disk:[dir.]
```

Define a Concealed, Rooted Logical

To define a concealed, rooted logical

1. Enter one of the following commands at the operating system prompt, as appropriate.

Group-level installation (while logged into an account in the appropriate group and the GRPNAM privileges):

```
$ define/group/exec/translation=concealed -  
    logicalname disk:[dir.]
```

System-level installation:

```
$ define/system/exec/translation=concealed -  
    logical name disk:[dir.]
```

where:

logicalname

Is the name you assign to the concealed logical representing the location for a particular set of files.

disk

Is the disk on which these files will reside.

dir

Is the directory specification, if appropriate, in which you want the install program to create the directory tree for storing the Ingres files.

2. Record the logical name for future reference.

Guidelines for VMS Cluster Configuration

Use the following guidelines when configuring a VMS cluster installation:

- All Ingres areas must be on commonly accessible cluster-mounted devices.
- Each node of a cluster installation must have its own transaction log file and backup transaction log file.
- All Ingres areas must be the same throughout the cluster, except for the areas for the transaction log file and backup transaction log file.
- Each node of a cluster installation can have its transaction log file and backup transaction log file in a separate location. However, the physical location of each node's transaction log files must be accessible to all nodes on the cluster that runs Ingres.

Installing and configuring Ingres in a VMS cluster environment is described in Ingres Cluster Solution for OpenVMS.

Owner of the System Administrator Account

The install program automatically creates the system administrator account if you want, after you specify its user name and user identification code (UIC).

Note: When the install program creates the system administrator account, it automatically sets up the minimum recommended process resources and privileges. However, if you provide the user name or UIC of an existing account, you must ensure that the account has the appropriate resources and privileges.

Each separate installation must be owned by a *unique* OpenVMS account, even if the same person acts as system administrator for all installations. This applies to both system-level and group-level installations.

For a system-level installation, the recommended user name for the system administrator account is `ingres`; for group-level installations, it is `ingresxx`, where `xx` is the installation ID. These user names are suggested, but not required. You can substitute different names.

Note: The system administrator account must not be the "SYSTEM" account. A separate, privileged account is required.

UIC for an Installation

You can use any user identification code (UIC) for the account that owns a system-level installation as long as it is not used by any other Ingres installation. For a group-level installation, we recommend that you choose a group UIC that does not automatically confer system privileges. This means a UIC greater than the `SYSGEN` parameter, `MAXSYSGROUP`.

If you have more than one installation on a particular node, you must place these installations in separate group UICs. For example, if the UIC for the system-level installation is `[300,1]`, the UIC for a group installation on the same node can be `[400,1]`, but cannot be `[300,*]`.

The UIC determines the group logical name table for a group-level installation. The users whose group UIC has an installation will be allowed to access their group's installation by default. Scripts are provided to switch between installations, but users can become confused. Therefore, it is best to be sure the UIC for a given group installation does not include users who must access the system-level installation or a different group installation.

Run the Setup Programs

The Setup programs configure installed components.

The Ingres Installation Verification Procedure (IVP) runs the Setup programs automatically at the end of the VMSINSTAL step, if you respond yes to the following installation program prompt:

```
Do you want to run the configuration IVP following this installation [YES] ?
```

If you choose to postpone running the setup programs or need to run them on another node in an Ingres cluster installation, you can invoke each setup program individually.

The order in which you execute the Setup programs is important. For a networked server installation, run the Setup program for the DBMS Server before running Ingres Net Setup. Ingres Net can then use several of the values you specified for the DBMS Server. Run Ingres Star Setup last.

To run a setup program

1. Enter the appropriate command, as follows:

```
$ @II_SYSTEM:[ingres.utility]program_name
```

where *program_name* is the name of the setup program, as follows:

iisudbms

DBMS Server setup

iisuc2

C2 Security auditing

iisunet

Ingres Net setup

iisudas

Data Access Server setup

iisubr

Ingres Bridge setup

iisuabf

Applications-By-Forms setup

iisurep

Ingres Replicator setup

iisuodbc

ODBC Driver setup

iisustar

Ingres Star setup

iisurms

RMS Gateway setup

iisukerberos

Kerberos setup

The Setup program for each component displays a description of its purpose and any other important information you should know before setting up the software.

2. Respond to the program prompts.

If you are setting up Ingres for clusters, you must provide a unique cluster identification number, between 1 and 32, which identifies the current node.

When setting up required parameters, default values sometimes appear in brackets after the prompt, as in the following example:

How many concurrent users do you want to support? [32]

To accept a default value, press Return. Otherwise, enter an appropriate value and press Return.

If you do not upgrade all databases during setup, you can upgrade specific databases later, using the `upgradedb` command (which also runs the utility that upgrades the tools catalogs). For more information, see the *Migration Guide*.

When the Setup programs complete, you are returned to the operating system prompt. Your Ingres installation is now configured.

Post-Installation Tasks

Post-installation tasks include the following:

- Add an automatic startup command
- Establish user access to tools and databases
- Maintain the installation history log file

How You Add an Automatic Startup Command

If you want your installation to start automatically whenever the system is rebooted, the operating system administrator must add the appropriate startup command to the system startup file.

Note: For any additions to this procedure that are necessary for your platform, see the Readme file.

To add an automatic startup command, follow these steps:

1. Log in to your system as the SYSTEM account.
2. Edit the system startup file. The install process places the appropriate startup command in the system startup file as a comment:

```
@SYS$STARTUP:INGRES_STARTUP.COM
```

Remove the comment status and delete any duplicate entries.

3. Edit the SYS\$STARTUP:INGRES_STARTUP.COM file to meet the needs of your environment.

How You Establish User Access to Tools and Databases

The install process identifies only the owner of the system administrator account to your installation to permit authorized access to the databases.

To enable users to access tools and databases, the system administrator must do the following:

- Edit the user login files to facilitate user access to the tools needed to query the databases.
- Use the accessdb command or CREATE USER statement to authorize users to access specific databases.
- Ensure that the appropriate logical names are “visible” to the users

Facilitate User Access to Tools

To facilitate user access to tools when they log in to the system, add the following access commands to the login.com file for the user:

For users, add:

```
@II_SYSTEM:[ingres]ingusrdef.com
```

For database administrators, add:

```
@II_SYSTEM:[ingres]ingdbadef.com
```

For the system administrator, add:

```
@II_SYSTEM:[ingres]ingsysdef.com
```

The commands in these files provide tools access to all users in a system-level installation and to all users with the appropriate group user identification code (UIC) in a group-level installation.

Installation History Log File

The install program records in the installation history log file each new installation as it occurs:

```
SYS$COMMON:[sysex]ingres_installations.dat
```

The installation history log file is placed in this location so that it can be as globally accessible as possible. VMSINSTAL will display its contents during installation to help avoid mistakes, so it is helpful to keep it accurate.

Edit the Installation History Log File

The system administrator can edit the installation history file to add data for an existing installation or to maintain data for your new installations.

To edit the installation history log file, use the following executable file provided with the distribution: II_SYSTEM:[ingres.install]ii_installs.com.

To run the ii_installs.com script

1. Log in as the system administrator and type the following at the system prompt:

```
$ @II_SYSTEM:[ingres.install]ii_installs
```

The II_INSTALLS> prompt appears.

2. Type the following:
help
3. Follow the instructions provided on your screen to view or add entries to this file.

Note: If editing the file, be sure to maintain the sequential order of wrapped rows.

Ingres Cluster Solution for OpenVMS

The Ingres Cluster Solution is a variation of a typical Ingres installation in which Ingres runs simultaneously on multiple host machines to provide cooperative and transparent access to one or more databases.

Ingres Cluster is a full server installation, except that file locations must be storage locations accessible from each node that is part of the cluster. Once installed, you run the `iimkcluster` utility to convert the initial installation into one of the cluster members (nodes), and then run the `iisunode` utility to add more nodes.

Use of the Ingres Cluster Solution is incompatible with the following Ingres features:

- Row-level locking
- Update mode locks
- Two-phase commit (2PC)
- Auditing of tables with long data types

On each node, make sure that Ingres and your applications perform as expected. Ingres internally handles some restrictions by converting some lock levels and lock modes to lower level locks and stronger lock modes, which may result in increased contention or deadlocks.

Requirements for the Ingres Cluster Solution on OpenVMS

The Ingres Cluster Solution has the following requirements:

- An Alpha VMS Cluster whose component machines are running VMS version 7.3-2 or above.
- DECnet available on all machines.
- Sufficient resources to run Ingres as a standard server installation on each machine.
- Sufficient storage accessible to all the machines in the cluster. The number and capacity of the shared storage devices should follow the guidelines specified for a standard Ingres server install.

How You Prepare to Install the Ingres Cluster Solution for OpenVMS

Before installing Ingres in a VMS Cluster environment, follow these steps:

1. Check the Ingres Technical Support web site for the latest cluster installation procedures and supported hardware and software. Download the required software.
2. Plan your storage location layout as described for a stand-alone Ingres installation, with the restriction that all your file locations—including data (II_DATABASE), transaction log (II_LOG_FILE, II_DUAL_LOG), checkpoint (II_CHECKPOINT), journal (II_JOURNAL), and dump (II_DUMP)—must reside on cluster mounted disks.

How You Install the Ingres Cluster Solution for OpenVMS

Follow these steps to install and configure Ingres in a VMS Cluster environment.

1. Install Ingres in a stand-alone configuration, using locations that reside on cluster mounted disks.
2. Verify that the stand-alone Ingres installation operates correctly.

Typically, it is easier to resolve any configuration issues at this stage because only one machine is in use.

3. When you are confident that the stand-alone Ingres is operating correctly, shut down the installation. As the user that owns the installation, execute the `iimkcluster` utility, as follows:

```
iimkcluster
```

The utility prompts you for a node number.

Node numbers are unique integers in the range 1 through the maximum supported cluster members for your platform (currently 15). During a partial cluster failure, the surviving cluster member (node) with the lowest node number is responsible for recovering transactions on behalf of the failed nodes, so you should assign low numbers to the more powerful machines in the cluster.

The `iimkcluster` utility renames the transaction logs and certain diagnostic log files (`iircp.log`, `iiacp.log`, and so on) by appending the host name of the machine on which the cluster member is running. Also created is a sub-directory in the `II_SYSTEM:[ingres.files.memory]` directory with the name of the host machine, and directory `II_SYSTEM:[ingres.admin.hostname]`, which is currently unused.

This step keeps entities that are normally operated upon by only one node separate from corresponding objects that will be created by the other nodes.

4. Restart Ingres. Confirm that all processes have started. Confirm the initial node is operational by performing a few sanity checks such as creating and destroying a scratch database.

You should also perform application testing to confirm that certain Ingres Cluster Solution restrictions, such as lack of support for row-level locking, will not impact the usability of your applications.

5. Shut down Ingres.

6. Run the `iisunode` utility on each node. As the user that owns the installation, enter the following:

```
iisunode
```

The utility prompts you for a unique node number. Once entered and confirmed, `iisunode` does the following:

- Adds the same directories for the new node as `iimkcluster` created for the initial node
 - Duplicates the configuration information from the initial node
 - Creates the transaction logs for this node
7. Start Ingres individually on each node, and verify correct operation.

How Client Applications Access an Ingres Cluster

Applications can access Ingres configured for Ingres Cluster Solution by using any of the following methods:

- Running directly on one of the cluster member machines
- Connecting directly to a specific cluster member using a `vnode` defined with the network address of the cluster member
- Selecting a connection to any available member of the cluster, or a subset thereof, by using a `vnode` defined with multiple network addresses.

Appendix C: Configuring the High Availability Option for Red Hat Cluster Suite

This section contains the following topics:

[What You Need to Know](#) (see page 101)

[A Database Service](#) (see page 101)

[How You Set Up a Cluster Service for an Ingres Database](#) (see page 102)

[Add an Ingres Service](#) (see page 104)

What You Need to Know

This appendix describes how to configure the Ingres High Availability Option for the Red Hat Cluster Suite. It assumes you have read *Configuring and Managing a Cluster* of the Red Hat Cluster Suite documentation and the *Ingres Installation Guide*.

A Database Service

A database service can serve highly-available data to an Ingres database application. The application can then provide network access to database client systems. If the service fails over, the application accesses the shared database data through the new cluster system. A network-accessible database service is usually assigned an IP address, which is failed over along with the service to maintain transparent access for clients.

How You Set Up a Cluster Service for an Ingres Database

This section provides an example of setting up a cluster service for an Ingres database. Although the variables used in the service scripts depend on the specific Ingres configuration, the example may aid in setting up a service for individual environments.

In the example that follows:

- The service includes one IP address for the Ingres clients to use.
- The service has two mounted file systems, one for the Ingres software (/gfs-a/) and the other for the Ingres database (/gfs-b/), which are set up before the service is added.
- A unique cluster name ingrescluster is attributed to the cluster. This allows shared Ingres software instances across the failover cluster.
- The Ingres instance ID for the installed packages is II.

To set up an Ingres service, follow these steps:

1. Export the cluster name before installing, then install the Ingres software. For example:

```
export II_HOSTNAME=ingrescluster
<Install according to the Ingres Installation Guide>
<copy /etc/init.d/ingres<INSTANCE> to other node>
<copy /home/ingres/.ing<INSTANCE>bash to other node>
```

2. The Ingres service example uses a wrapper script, `ha_ingres`, which is used to start, stop, and query the Ingres service. This script calls the Ingres init scripts installed above. The `ha_ingres` script must be placed in `/etc/init.d`, owned by root and executable (`chmod +x`). Specify this script when you add the service.

Here is an example of the `ha_ingres` script:

```
#!/bin/sh
#
# Cluster service script to start, stop, and check status of Ingres
#

export II_HOSTNAME=ingrescluster
export II_INSTANCE=II

case $1 in
  status)
    . /home/ingres/.ing${II_INSTANCE}bash
    echo "\q"|sql iidbdb &> /dev/null
    rc=$?
    test $rc -eq 0 && echo "Running" || echo "Not Running"
    ;;
  *)
    /etc/init.d/ingres${II_INSTANCE} $@
    rc=$?
    ;;
esac

exit $rc
```

Add an Ingres Service

After the cluster service for the Ingres database is set up, you must add it using the Cluster Configuration tool.

Follow these steps:

1. Type **system-config-cluster** at a shell prompt.
The Cluster Configuration tool is started.
2. Add the Ingres service by using the controls in the Cluster Configuration tool.
 - a. Enter a Service Name for the Ingres service (that is, ha_ingres).
 - b. Select a Failover Domain or leave it as None (see Red Hat Cluster documentation).
 - c. Select other options, such as a recovery policy, and whether to autostart this service.
 - d. Specify the ha_ingres script (defined earlier) as a resource for the Ingres service.
 - e. (Optional) Add an IP address for the Ingres service (only if you want to host a virtual IP address on the cluster).
3. Make sure the service is added to all nodes in the cluster.

Appendix D: Configuring the High Availability Option for Windows Clustering

This section contains the following topics:

[What You Need to Know](#) (see page 105)

[Ingres High Availability Option](#) (see page 106)

[Failover](#) (see page 106)

[Configuration Guidelines for System and Data Files](#) (see page 106)

[DBMS Server Requirements](#) (see page 107)

[Windows Cluster Service Installation](#) (see page 108)

[Prerequisites for High Availability Option](#) (see page 109)

[Install High Availability Option for Windows Clustering](#) (see page 110)

[Starting and Stopping Ingres in a Cluster Environment](#) (see page 111)

[How the High Availability Option Is Removed](#) (see page 113)

What You Need to Know

This appendix describes how to install and configure the Ingres High Availability Option in Windows cluster environments.

The information in this appendix is intended for engineers with extensive knowledge of software and hardware in the cluster environment, including cluster concepts such as high availability (HA), cluster disk, resource types, virtual servers, cluster administrator, and failover.

Note: This appendix assumes knowledge of Microsoft Windows Server, Windows Clustering, and Windows Cluster Service. It refers to specific Windows Cluster Service commands and procedures. For additional information about these commands, see your Windows Cluster Service documentation.

Ingres High Availability Option

The Ingres High Availability Option for clusters runs application instances only on a single node. Depending on how the data service has been configured, when the fault monitor detects an error, it either attempts to restart the instance on the same node or to start the instance on another node for failover.

Note: The Ingres High Availability Option is not scalable; that is, it does not provide active instances on multiple nodes.

Failover

Failover is the process by which a cluster automatically relocates a data service from a failed primary node to a designated secondary node. If a failover occurs, a brief interruption in service can occur, requiring you to reconnect after the failover has completed. Once reconnected, the physical server on which Ingres is running is transparent to the end user.

The automatic failover feature ensures that transactions are handled by an active node in the cluster if a hardware or software failure occurs.

Note: Any transactions that are in flight at the time of failure are lost.

Configuration Guidelines for System and Data Files

Ingres system files, data, and related files must exist on a cluster disk.

Cluster Disk

A cluster disk is a disk on a shared bus connected to the cluster nodes, which all the cluster nodes can access (though not at the same time).

Access to Ingres data does not depend on the type of cluster disk system (shared external SCSI, SCSI RAID, or SCSI over Fiber Channel).

Physical Disk Resource Type

The cluster service uses the Physical Disk resource type to manage disks that are on a cluster storage device. Disks must have the same drive letters on all cluster nodes.

To prevent data corruption, only one node at a time can use a cluster disk. Cluster Service normally prevents more than one node from using a disk at the same time. However, there are some circumstances not protected by ordinary cluster safeguards: when you install a new disk, and when you remove a disk from a cluster. For more information, see the Windows 2000 Server documentation.

The Ingres system, data, and related files must exist on a Physical Disk resource type.

Generic Service Resource Type

You use the Generic Service resource to manage Windows services as cluster resources.

The Ingres High Availability Option for Windows installs the Ingres Service as a Generic Service resource type. You must specify, during setup, the Physical Disk resources that contain the Ingres data and related files as dependencies of the Ingres Service cluster resource.

DBMS Server Requirements

When using the Ingres High Availability Option, the Ingres DBMS server is installed and configured as a typical installation, provided that the system files and database files are installed on the cluster file system.

For a clustered DBMS installation, the database, checkpoint, journal, dump, work, and transaction log files must be on a cluster drive. This configuration should be done while at the Ingres DBMS Configuration screen during installation.

Virtual Servers

To access a DBMS server in a non-cluster environment, network clients must connect to a physical server, identified by a network name and IP address. If that server fails, access to the DBMS is impossible.

Windows Clustering enables the creation of virtual servers, identified by a unique network name and IP address (different from the underlying physical server's network name and IP address). Unlike a physical server, a virtual server is not associated with a specific computer. If the node hosting the virtual server fails, clients can still access the DBMS using the same server name.

Virtual Server Name

Installing the Ingres High Availability Option modifies the Ingres installation to use the virtual server name. This changes the config.dat file by replacing the server name with the virtual server name and sets II_HOSTNAME to the virtual server name. To automatically reconnect to the server after a failover, network clients must use this virtual server name when connecting to the DBMS Server.

Windows Cluster Service Installation

The Windows Cluster Service must be installed and configured on the cluster nodes before installing the Ingres High Availability Option. The Cluster Service is a component of Windows Clustering that creates a server cluster and controls all aspects of its operation.

Cluster Administrator

Cluster Administrator is an application used to configure a cluster. It can run on any member of the trusted domain, whether the computer is a node of the cluster or not.

Verify Windows Cluster Service Installation

To verify the Windows Cluster Service installation, a system administrator can start Cluster Administrator from the Windows Start menu. Choose Start, Programs, Administrative Tools, Cluster Administrator.

Cluster Command

As an alternative to using Cluster Administrator to verify the Windows Cluster Service installation, you can use the cluster command from the Windows command prompt. Also, the command (cluster.exe) can be called from command scripts.

To verify the installation, a system administrator can use the following commands:

Command	Description
cluster /version	Displays the cluster version and service pack information.
cluster node	Displays the status of all nodes in the cluster.
cluster resource	Displays the status of all cluster resources.

Prerequisites for High Availability Option

The following are the prerequisites for installing Ingres High Availability Option on Windows:

- Ingres High Availability option supports Windows 2003 Enterprise Edition or Windows Server 2008.
- The Ingres directory (II_SYSTEM) must be installed on a local non-cluster drive.
- For a clustered DBMS, the following DBMS related files must be available on a cluster file system:
 - Database
 - Checkpoint
 - Journal
 - Dump
 - Work
- Ingres must be installed as a service.

Install High Availability Option for Windows Clustering

Installation of Ingres High Availability Option is an intermediary step during the Ingres installation. In the installation wizard, the Ingres Configuration dialog contains a Cluster Configuration section. The Cluster Configuration section consists of a checkbox for Install Ingres High Availability option and a text box for Ingres Cluster Resource Name. These options are only available on machines that are part of the cluster. They are disabled on machines that are not part of the cluster.

Functions Performed by the Ingres High Availability Option Setup Utility

The Ingres High Availability Option setup utility performs the following functions:

- Ensures that Ingres is installed as a service on the cluster node
- Registers the Ingres Service as a Generic Service cluster resource
- Creates or updates the cluster group for the DBMS Server or Communications Server (client-only)
- Establishes dependencies of the Ingres Service on the required cluster resources (network name and disk)
- Modifies the existing Ingres installation for Windows Clustering

All administration must be done as a system administrator.

Installation on Windows Server 2008

Using parallel accessed SCSI disks as cluster storage, while supported on Windows Server 2003, should not be used on Windows Server 2008. Make sure you have a NAS, SAN, or iSCSI storage solution before upgrading your Windows Server 2003 Failover Cluster to Windows Server 2008.

If you use an iSCSI attached storage device, problems can occur with the disk arbitration during Ingres installation. Restarting the cluster service should solve them, in most cases. For example, if Ingres installation fails during creation of the log file or the system databases, restart the cluster service by running **net stop clussvc** and **net start clussvc**.

Do not start multiple nodes at the same time. There should be a 30-second delay between the startup of each node.

Starting and Stopping Ingres in a Cluster Environment

When using Ingres in a cluster environment, it must be started and stopped through the cluster service.

If Ingres is shut down while under the control of the High Availability Option, the cluster service sees this as a failure and either tries to restart Ingres on the same node or fail over the service to the failover node.

Start Ingres in a Cluster Environment

Use the Windows Cluster Administrator or the cluster command to bring Ingres online in a cluster environment.

Note: Do **not** use the `ingstart` command to start Ingres in a cluster environment.

To bring Ingres online in a cluster environment using the cluster command

```
C:> cluster resource <ingres_resource_name> /online /wait:300
```

To bring the entire Ingres group online, use the following command:

```
C:> cluster group <ingres_resource_group> /online /wait:300
```

If you omit `"/wait:300,"` you may see the following message, which indicates that the Ingres service did not reply within the default time period. However, the cluster command executed properly.

```
System error 997 has occurred.  
Overlapped I/O operation is in progress.
```

Stop Ingres in a Cluster Environment

Use the Windows Cluster Administrator or the cluster command to take Ingres offline in a cluster environment.

Note: Do **not** use the `ingstop` command to stop Ingres in a cluster environment.

To take Ingres offline using the cluster command

```
C:> cluster resource <ingres_resource_name> /offline /wait:300
```

To take the entire Ingres group offline, use the following command:

```
C:> cluster group <ingres_resource_group> /offline /wait:300
```

If you omit `/wait:300,` you may see the following message, which indicates that the Ingres service did not reply within the default time period. However, the cluster command executed properly.

```
System error 997 has occurred.  
Overlapped I/O operation is in progress.
```

How the High Availability Option Is Removed

Use the Windows Cluster Administrator or the cluster command to remove the Ingres High Availability Option.

Follow these basic steps:

1. Take the Ingres resource offline.
2. Move the resource group containing the Ingres service to the node where you originally installed Ingres.
3. Uninstall Ingres on the node where you originally installed it.
4. Delete the resource.

To remove the Ingres High Availability Option using the cluster command

1. Take Ingres offline:

```
C:> cluster resource <ingres_resource_name> /offline /wait:300
```

If you omit `"/wait:300,"` you may see the following message, which indicates that the Ingres service did not reply within the default time period. However, the cluster command executed properly.

```
System error 997 has occurred.  
Overlapped I/O operation is in progress.
```

2. Move the resource group containing the Ingres service to the node where you originally installed Ingres.

```
C:> cluster group <ingres_group_name> /move:<node_name>
```

3. Uninstall Ingres on the node where you originally installed it.
4. Delete the resource using the following cluster command:

```
C:> cluster resource <ingres_resource_name> /delete
```


Appendix E: Configuring the High Availability Option for Sun

This section contains the following topics:

[What You Need to Know](#) (see page 115)

[Failover](#) (see page 116)

[Configuration Guidelines for System and Data Files](#) (see page 116)

[DBMS Server Requirements](#) (see page 116)

[Sun Cluster Software Installation Verification](#) (see page 117)

[How You Install the Ingres High Availability Option for Sun](#) (see page 117)

[How the Ingres High Availability Option Is Configured](#) (see page 120)

[How the Database Probe Function Is Configured](#) (see page 121)

[Turn on Debugging and Tracing for the High Availability Option](#) (see page 122)

[Starting and Stopping Ingres in a Cluster Environment](#) (see page 122)

[Clear a Fault in a Resource Group](#) (see page 123)

[How You Remove the High Availability Option](#) (see page 124)

What You Need to Know

This appendix describes how to install and configure the Ingres High Availability Option in Sun cluster environments.

The information in this appendix is intended for engineers with extensive knowledge of software and hardware in the cluster environment, including cluster concepts such as high availability (HA), global file systems, the resource management object model, resource properties, logical host names, shared addresses, failover, and switchover.

Note: This appendix refers to specific Sun Cluster administration commands. For additional information about these commands, consult your Sun Cluster documentation.

Failover

Failover is the process by which a cluster automatically relocates a data service from a failed primary node to a designated secondary node. If a failover occurs, a brief interruption in service can occur, requiring you to reconnect after the failover has completed. Once reconnected, the physical server on which Ingres is running is transparent to the end user.

The automatic failover feature ensures that transactions are handled by an active node in the cluster if a hardware or software failure occurs.

Note: Any transactions that are in flight at the time of failure are lost.

Configuration Guidelines for System and Data Files

Ingres system files can exist on either the local disks of each cluster node or on the cluster file system.

Note: The provided `install.sh` script assumes that the Ingres system files will be installed on the cluster file system.

Ingres database files must also exist on the shared cluster file system.

Cluster File Systems

Access to Ingres data does not depend on the type of cluster file system. The Ingres HA Option install assumes that you have already defined a cluster file system.

DBMS Server Requirements

When using the Ingres High Availability Option, the Ingres DBMS server is installed and configured as a typical installation, provided that the system files and database files are installed on the cluster file system.

For a clustered DBMS installation, the database, checkpoint, journal, dump, work, and transaction log files must be on a cluster drive. This configuration should be done while at the Ingres DBMS Configuration screen during installation.

Sun Cluster Software Installation Verification

The Sun Cluster packages must be installed and configured on the cluster nodes before installing the Ingres High Availability Option. The installation must be verified by a qualified Sun Cluster Engineer using the commands shown in the following table:

Command	Description
scinstall -pv	Displays the Sun cluster release and version information.
scrgadm -p	Displays the Sun cluster configured resource types, resource groups, and resources.
scstat -p	Displays the status of the Sun cluster components.
scconf -p	Displays the cluster configuration.

How You Install the Ingres High Availability Option for Sun

The Ingres High Availability Option for Sun is distributed as a separate package named IngresSCAgent that can be downloaded from the Ingres web site.

Follow this process to install the High Availability Option:

1. Download the IngresSCAgent package and extract its contents into a temporary directory (for example, IngresSCAgent-9.2.0.130-sun-solaris-sparc-32-64bit).
2. Log in as root to continue the installation.
3. Source the `$II_SYSTEM/ingres/.ingIIsh` script to set up your environment with the correct `$II_SYSTEM` and `$LD_LIBRARY_PATH` settings.

```
# . $II_SYSTEM/ingres/.ingIIsh  
# ./install.sh
```
4. Execute the `install.sh` script as root from the IngresSCAgent package to begin the installation.

How to Respond to Installation Script Prompts

The High Availability Option installation script prompts you to complete the following steps:

1. Install Ingres High Availability Option package files.

The Ingres High Availability Option consists of an Ingres resource type and runtime scripts, which are installed as part of the High Availability option package files install. The High Availability option package files must be accessible from every node that Ingres can fail over. The package files must be installed either on a cluster file system or on all nodes in the cluster where the Ingres DBMS Server can fail over.

The `install.sh` script installs the High Availability Option package files by default in local directory `/opt/Ingres/IngresSCAgent`. If you choose to install these files on a local file system, you should execute this install step on every failover node before continuing to the next step in the install process.

The resource file `Ingres.ingres_server` is copied to the standard resource type directory `/usr/cluster/lib/rgm/rtreg`.

Note: When installing the HA option files on a local file system, the actual path should be identical on each node.

2. Install Ingres.ingres_server resource type.

This step registers the Ingres resource type (Ingres.ingres_server). Resource type registration is required only once on the cluster.

Multiple Ingres failover installations can be installed in separate failover groups, but they will all share the same Ingres resource type.

3. Configure Ingres installation.

This step modifies the Ingres installation configuration options to create a logical hostname and then renames the configuration options in config.dat.

This step ensures that Ingres has the same logical hostname and reads the same logical configuration options when run on any failover node.

4. Create resource group containing an Ingres installation.

This step creates an Ingres resource group and a resource in the group that corresponds to the Ingres installation.

The install script generates a default group name, but you can override this during installation. By default, this group will contain all cluster nodes as potential masters.

The install script generates a default resource name for the Ingres installation, but you can override this resource name during installation.

5. Add the cluster logical hostname to the Ingres resource group.

This step creates a logical hostname resource and adds it to the Ingres resource group. A logical hostname allows a floating IP address to be defined that moves from one node to another, along with the Ingres installation, during failover. This ensures that one IP address and DNS entry can be used to reference the Ingres installation regardless of the node it is running on.

This step prompts for a logical hostname, which can be entered as a DNS hostname or as a static IP address.

How the Ingres High Availability Option Is Configured

During the installation of the Ingres High Availability Option, several resource variables are defined to control the operation of the HA Option scripts. These variables define:

- The location and user ID for the Ingres installation
- The tracing flags to be used for debugging
- The values needed to enable the database probe function

Some of these variables can be changed after installation to change the operation of the Ingres failover scripts.

Examine the following scripts in IngresSCAgent/bin to determine exactly how these variables are used. Examine the Sun Cluster documentation to determine when these scripts are called.

- `ingres_monitor_start`
- `ingres_monitor_stop`
- `ingres_probe`
- `ingres_server_start`
- `ingres_server_stop`
- `ingres_update`
- `ingres_validate`

Use the following Sun Cluster administration command syntax to change the value of a resource variable:

```
# scrgadm -c -j resource-name -x variable=value
```

The following is a list of variables that can be defined for the Ingres High Availability Option scripts.

II_SYSTEM=*path*

Sets the path for the Ingres system directory.

II_OWNER=*userid*

Sets the Ingres owner.

II_HOSTNAME=*logical_host*

Sets the logical host name.

User=*userid* (Optional)

Sets the users for secondary monitoring.

Database=*database* (Optional)

Sets the database for secondary monitoring.

Table=*database_table* (Optional)

Sets the database table for secondary monitoring.

Probe_timeout=*seconds* (Optional)

Sets the timeout value for the probe database call.

EnvFile=*path* (Optional)

Sets the file used to load additional user specified environment variables.

II_SCHA_TRACE=*n* (Optional)

Sets the trace level for debugging. Valid levels are 0 for no tracing and 1-9 for debugging information.

II_SCHA_LOG=*path* (Optional)

Sets the file name for the trace output.

How the Database Probe Function Is Configured

A feature of the Solaris Cluster software, the database Probe function allows the Cluster to verify that an application is still working correctly and still accessible.

To determine how the probe script operates, examine the IngresSCAgent/bin/ingres_probe file.

To enable the Ingres High Availability Option database probe, you must define the variables User, Database, Table, and Probe_timeout.

An optional variable Envfile specifies an external script that can be used to set up additional script variables during the database probe.

To use the ingres_probe feature you must first create a database with a table that contains the column "cluster_monitor". The owner of the database and table must also be defined if it does not already exist.

The following commands are an example of how to create the necessary database and table:

```
$ sql iidbdb
* CREATE USER User \g
* COMMIT \g \q

$ sql Database -uUser
* CREATE TABLE User.Table ( cluster_monitor DATE ) ; \g
* INSERT INTO User.Table (cluster_monitor) VALUES (DATE('NOW')); \g
* COMMIT \g \q
```

Turn on Debugging and Tracing for the High Availability Option

Debugging and tracing of the Ingres High Availability Option for Sun is controlled by the II_SCHA_LOG and II_SCHA_TRACE variables.

To turn on debugging

Set the II_SCHA_TRACE level and II_SCHA_LOG file name using the following commands:

```
# scgradm -c -j ingres_resource_name -x II_SCHA_LOG=path_to_debug_output  
# scrgadm -c -j ingres_resource_name -x II_SCHA_TRACE=<0-9>
```

where:

0=none, 1=minimal, and 9=maximum.

To turn off debugging

Set the level to 0 and the path to "", as follows:

```
# scgradm -c -j ingres_resource_name -x II_SCHA_LOG=""  
# scgradm -c -j ingres_resource_name -x II_SCHA_TRACE=0
```

Starting and Stopping Ingres in a Cluster Environment

When using Ingres in a cluster environment, it must be started and stopped through the cluster service.

If Ingres is shut down while under the control of the High Availability Option, the cluster service sees this as a failure and either tries to restart Ingres on the same node or fail over the service to the failover node.

Start Ingres in a Cluster Environment

To bring Ingres online in a cluster environment, use the following Sun Cluster command:

```
# scswitch -Z -g ingres_resource_group
```

Note: Do **not** use the `ingstart` command to start Ingres in a cluster environment.

Stop Ingres in a Cluster Environment

To bring Ingres offline, use the following Sun Cluster command:

```
# scswitch -F -g ingres_resource_group
```

Note: Do **not** use the `ingstop` command to stop Ingres in a cluster environment.

Clear a Fault in a Resource Group

If the resource group has a fault, Ingres cannot start.

To clear the failure, use the following Sun Cluster command:

```
# scswitch -c -h node -j ingres_resource_name -f STOP_FAILED
```

To make sure Ingres is shut down, use the command:

```
# scswitch -F -g ingres_resource_group
```

How You Remove the High Availability Option

The Ingres High Availability option installs a simple failover configuration which most likely will be re-configured to be more complicated in a production environment. An automated uninstall process is not provided because it would be difficult to anticipate all possible failover configurations.

Use these steps only as a guideline to the process of uninstalling the High Availability Option:

1. Bring Ingres offline using the following Sun Cluster command:

```
# scswitch -F -g ingres_resource_group
```
2. Disable the resource using the following Sun Cluster command:

```
# scswitch -n -j ingres_resource_name
```
3. Remove the resource using the following Sun Cluster command:

```
# scrgadm -r -j ingres_resource_name
```
4. Disable the logical host using the following Sun Cluster command:

```
# scswitch -n -j logical_hostname
```
5. Remove the logical host using the following Sun Cluster command:

```
# scrgadm -r -j logical_hostname
```
6. Remove the resource group using the following Sun Cluster command:

```
# scrgadm -r -g ingres_resource_group
```
7. Remove the resource type using the following Sun Cluster command:

```
# scrgadm -r -t Ingres.ingres_server
```
8. Remove the Ingres.ingres_server resource type file under /usr/cluster/lib/rgm/rtreg on each failover node.
9. Remove the IngresSCAgent directory and files installed on each failover node. The default directory is /opt/Ingres/IngresSCAgent.

Appendix F: System Requirements for OpenVMS

This section contains the following topics:

[System Hardware Requirements on OpenVMS](#) (see page 125)

[SYSGEN Parameters that Affect Operation](#) (see page 126)

[Recommended OpenVMS Process Resources](#) (see page 129)

[Required OpenVMS Privileges](#) (see page 131)

Note: The recommended values mentioned in this appendix are for a minimal Ingres installation. Requirements for your system may vary, depending on the products you install and the maximum number of users. For values that may be better suited to your environment, see the Ingres technical support page, accessible from <http://ingres.com>.

System Hardware Requirements on OpenVMS

The following table describes typical system hardware requirements.

Note: Check the Readme file for changes to the following requirements that are specific to your platform.

Element	Typical Requirement						
Number of disks	At least two separate storage devices for your databases and checkpoint files are strongly recommended.						
Disk space	<p>The amount of space required on any one particular disk is determined by the number of disks in your configuration, the packages you are installing, and the locations you choose for your files.</p> <p>Peak system disk utilization is approximately 1.2 x kit size.</p> <p>The default file size for your primary and backup transaction log files is 256 MB. However, the recommended size is 400 MB or larger.</p> <p>For help in determining space requirements for II_DATABASE, see the <i>Database Administrator Guide</i>.</p>						
Physical Memory	<p>For a DBMS Server installation:</p> <table border="1"><thead><tr><th>OS</th><th>Required</th><th>Recommended</th></tr></thead><tbody><tr><td>Alpha</td><td>128 MB</td><td>512 MB or more</td></tr></tbody></table>	OS	Required	Recommended	Alpha	128 MB	512 MB or more
OS	Required	Recommended					
Alpha	128 MB	512 MB or more					

SYSGEN Parameters that Affect Operation

It is important to understand the way OpenVMS parameter settings affect Ingres.

For information on tuning OpenVMS, see your OpenVMS documentation.

Note: Check the Readme file for changes to the following requirements that are specific to your platform.

The following table lists SYSGEN parameters that affect Ingres operation:

Element	Description	Considerations
CHANNELCNT	Number of open I/O channels permitted each OpenVMS process at any one time	<p>This parameter is critical to the successful execution of the DBMS Server. If the setting is too large, it wastes system memory; if too small, it limits Ingres I/O capabilities. Since the DBMS is multi-threaded, it effectively requires the same CHANNELCNT setting as if it were several individual processes.</p> <p>Use the following formula to calculate an approximate value, where <i>sessions</i> is the total number of connected sessions and <i>file_limit</i> is the total number of files that can be open at one time:</p> $\text{CHANNELCNT} = (\textit{sessions} * 4) + \textit{file_limit}$ <p>In general, set CHANNELCNT to a very large number.</p> <p>The <i>file_limit</i> (fillm) requirements will vary over time. The DBMS Server will, at most, open every file in every database, plus approximately five for the transaction log and error log files.</p>

Element	Description	Considerations
GBLSECTIONS	<p>Determines the maximum number of global sections that can be made known to the system by allocating the necessary storage for the Global Storage Table (GST) entries</p> <p>Example value for Alpha: 800</p>	<p>Several Ingres components use global pages of memory to enable multiple processes to share data.</p> <p>Examples are:</p> <ul style="list-style-type: none"> ■ Fast message storage that keeps an in-memory copy of Ingres message files ■ Multiple DBMS Servers configured to share in-memory caching of Ingres data pages
GBLPAGES	<p>Determines the size of the global page table and the limit for the total number of global pages that can be created</p> <p>Example value for Alpha: 250,000</p>	<ul style="list-style-type: none"> ■ Logging and locking system shared memory, which is allocated from global pages at startup time (the size of which is determined by the configuration of the logging and locking system)
GBLPAGFIL	<p>Establishes the maximum number of global pages with page file "Backing Storage Address" storage that can be created</p> <p>Example Value for Alpha: 40,000</p>	<p>On Alpha, memory pages are at least 8 KB in size. At this size, four Ingres pages fit in each Alpha memory page. Note that Alpha pages can be from 8 KB to 16 KB in size. See your OpenVMS Alpha documentation.</p> <p>To examine entries and available pages and sections, use the INSTALL utility and issue LIST/GLOBAL and LIST/GLOBAL/SUMMARY.</p>
NPAGEDYN	<p>Non-paged dynamic pool; determines the number of bytes to allocate for the non-pages dynamic pool</p>	<p>Used in allocating the logging and locking system in previous Ingres releases.</p> <p>Ingres uses GBLPAGES (global pages) rather than NPAGEDYN and NPAGEVIR to allocate the logging and locking system shared memory at startup time.</p>
NPAGEVIR	<p>Virtual non-paged pool; determines the number of bytes to which the non-paged pool may be extended</p>	<p>Used in allocating the logging and locking system in previous Ingres releases.</p> <p>Ingres uses GBLPAGES (global pages) rather than NPAGEDYN and NPAGEVIR to allocate the logging and locking system shared memory at startup time.</p>

Element	Description	Considerations
PQL_*	PQL_parameters; PQL_D is the default and PQL_M is the minimum process quotas that OpenVMS uses when creating detached processes (of which Ingres generates several)	<p>If a value is not specified on the command line when Ingres processes are started, the PQL default is used. If the process specifies a value below the PQL minimum, the current minimum is used. If the creator has DETACH privilege, any of the values can be overridden, which is crucial to the successful startup of Ingres processes.</p> <p>To examine the resulting process configuration, use:</p> <p>SHOW PROCESS/QUOTA/ID=pid</p>
VIRTUALPAGECNT	<p>Maximum virtual page count; determines the total number of pages that can be mapped for a process, which can be divided in any fashion between P0 and P1 space</p> <p>Beginning with OpenVMS 7.0, this parameter ceased to be tunable on Alpha as the process page tables have migrated from system space to a dedicated page table space. See your OpenVMS documentation for details.</p>	<p>The DBMS Server process can grow to require large numbers of virtual pages due to the complexity of its facilities and their various caches. It is critical that this process be able to expand its memory regions as necessary.</p> <p>If an error occurs while trying to expand the virtual size of the server, first examine the process page file quota, and then examine this system-wide limit. To do so, set the process page file quota to the current VIRTUALPAGECNT and start the server. If this does not resolve the problem, this parameter needs to be adjusted upwards.</p> <p>To examine current usage, use: SHOW PROCESS/QUOTA/ID=pid</p>
WSMAX	<p>Maximum size of process working set; determines the system-wide maximum size of a process working set, regardless of process quota</p> <p>Example value for Alpha: 130,000</p>	<p>This parameter limits the size of the physical memory that any OpenVMS process can grow to consume. To prevent Ingres processes from spending overhead page faulting, be sure to configure them with a working set large enough to reduce or eliminate the amount of swapping and/or paging required.</p> <p>Distinguish hard faults (disk reads/writes, which are expensive) from soft faults (memory read/writes, which are more tolerable).</p>

Recommended OpenVMS Process Resources

Check and set the OpenVMS process resources, if necessary, for the following accounts:

- System administrator account
- User accounts
- Database administrator accounts

Almost all OpenVMS process quotas impart a cost only when a given resource is requested and consumed. If you limit the Ingres process resources, it may function, but its performance may be impaired. Therefore, we generally recommend high resource quotas for the system administrator account. If performance is a critical issue, try setting the quota to infinity, if permitted.

The following table shows the recommended minimum process resources for the system administrator. When the install program creates the system administrator account, it automatically sets the process resources to the recommended minimums shown in the table.

Database administrators who checkpoint their own databases (as opposed to the system administrator) indirectly use OpenVMS backup. You should set UAF parameters for their accounts. For details, see the OpenVMS backup manual.

UAF Name	Description	Recommended Starting Value
BYTLM	Buffered I/O byte count limit	500,000
TQELM	Timer queue entry limit	100
PGFLQUO	Paging file quota (possibly set to the system virtual page count)	256,000
DIOLM	Direct I/O count limit	200
BIOLM	Buffered I/O limit	400
FILLM	Open file limit	1000
PRCLM	Subprocess creation limit	15 or larger - Alpha
ASTLM	AST queue limit	200
ENQLM	Enqueue limit (non-cluster)	4000
JTQUOTA	Job logical name table byte limit	20,000
MAXDETACH	Maximum detached	0 (infinity)

UAF Name	Description	Recommended Starting Value
	processes limit	
PRIV	Privileges	ALTPRI, CMKRNL, IMPERSONATE, EXQUOTA, OPER, PRMGBL, PRMMBX, READALL, SHARE, SYSGBL, SYSLCK, SYSNAM, SYSPRV, TMPMBX, WORLD
DEVICE DIRECTORY	Device and directory for default login	II_SYSTEM:[Ingres]
DEFPRIV	Default privileges	ALTPRI, CMKRNL, IMPERSONATE, EXQUOTA, OPER, PRMGBL, PRMMBX, READALL, SHARE, SYSGBL, SYSLCK, SYSNAM, SYSPRV, TMPMBX, WORLD

OpenVMS Quotas and Ingres Parameters

The following table lists the Ingres configuration (config.dat) parameters and their equivalent OpenVMS quotas specified in the system service call used to create the various Ingres servers.

Ingres Configuration Parameter	PQL\$_limit
vms_ast_limit	ASTLM
vms_buffer_limit	BYTLM
vms_extent	WSEXTENT
vms_file_limit	FILLM
vms_io_buffered	BIOLM
vms_io_direct	DIOLM
vms_maximum_working_set	WSQUOTA
vms_page_file	PGFLQUOTA
vms_queue_limit	TQELM
vms_enqueue_limit	ENQLM
vms_working_set	WSDEFAULT

The `vms_privileges` parameter corresponds to the `privadr` parameter of `SYS$CREPRC`.

Required OpenVMS Privileges

The system administrator account must have the following OpenVMS privileges:

- ALTPRI
- CMKRNL
- EXQUOTA
- IMPERSONATE
- OPER
- NETMBX
- PRMGBL
- PRMMBX
- READALL
- SHARE
- SYSGBL
- SYSLCK
- SYSNAM
- SYSPRV
- TMPMBX
- WORLD

If the install program creates the system administrator account, it automatically assigns these privileges.

Note: If you want to turn off the `vms_accounting` parameters, which allow the Ingres processes to be started without a record in the VMS accounting file, the system administrator will also need the ACNT privilege. (If this is the case, all processes started by the administrator will not be in the VMS accounting audit trail).

Appendix G: Response File Parameters

This section contains the following topics:

[II_RESPONSE_FILE](#) (see page 133)

[Response File—Define Configuration for Ingres Instance](#) (see page 133)

[List of Response File Parameters](#) (see page 134)

[Example Response File for Linux](#) (see page 136)

[Example Response File for Windows](#) (see page 136)

[Response File Parameters](#) (see page 138)

II_RESPONSE_FILE

Valid on: Windows, UNIX, Linux

II_RESPONSE_FILE is an Ingres environment variable that can point to the name of a valid response file. A response file is used to install a custom configuration of Ingres from the command line, to install Ingres silently, or to install embedded Ingres.

Response File—Define Configuration for Ingres Instance

A response file contains parameters that define how an Ingres instance is to be installed and configured. You can use any name for the response file.

Each entry in the response file must be on a separate line and in the following form:

variable=value

where *variable* is an Ingres response file parameter name.

List of Response File Parameters

The following lists, in alphabetical order, parameters that can be used in a response file.

Note: Some parameters are valid on certain operating systems only.

II_ADD_REMOVE_PROGRAMS (Windows)

II_ADD_TO_PATH (Windows)

II_CHARSET

II_CHECKPOINT

II_COMPONENT_CORE (Windows)

II_COMPONENT_DBMS (Windows)

II_COMPONENT_DOCUMENTATION (Windows)

II_COMPONENT_DOTNET (Windows)

II_COMPONENT_FRONTTOOLS (Windows)

II_COMPONENT_JDBC_CLIENT (Windows)

II_COMPONENT_NET (Windows)

II_COMPONENT_ODBC (Windows)

II_COMPONENT_REPLICATOR (Windows)

II_COMPONENT_SPATIAL (Windows)

II_COMPONENT_STAR (Windows)

II_COMPONENT_VISION (Windows)

II_CONFIG_TYPE

II_CREATE_FOLDER (Linux)

II_DATABASE

II_DATE_TYPE_ALIAS

II_DATE_FORMAT (Windows, Linux)

II_DEMODB_CREATE

II_DESTROY_TXLOG (Windows)

II_DUAL_LOG

II_DUMP

II_ENABLE_NETBIOS (Windows)

II_ENABLE_SPX (Windows)

II_ENABLE_SQL92

II_ENABLE_TCPIP (Windows)
II_GROUPLD (UNIX, Linux)
II_INSTALL_ALL_ICONS (Windows)
II_INSTALLATION
II_JOURNAL
II_LOCATION_DOCUMENTATION (Windows)
II_LOCATION_DOTNET (Windows)
II_LOG_FILE
II_LOG_FILE_SIZE_MB
II_MONEY_FORMAT (Windows, Linux)
II_REMOVE_ALL_FILES (Linux)
II_SERVICE_START_AUTO (Windows)
II_SERVICE_START_USER (Windows)
II_SERVICE_START_USERPASSWORD (Windows)
II_START_INGRES_ON_COMPLETE (Windows)
II_START_IVM_ON_COMPLETE (Windows)
II_START_ON_BOOT (Linux)
II_SYSTEM (Windows, UNIX)
II_TERMINAL (Windows, Linux)
II_TIMEZONE_NAME
II_UPGRADE_USER_DB
II_USERID (UNIX, Linux)
II_WORK

Example Response File for Linux

The following example response file includes all parameter options and their defaults for a Linux installation.

```
# Ingres Response File
# Generated by the Ingres Response File API
# Created: Thu Jan 25 13:14:24 2007

# [ Ingres Locations ]

II_DATABASE="/opt/Ingres/IngresII/"
II_CHECKPOINT="/opt/Ingres/IngresII/"
II_JOURNAL="/opt/Ingres/IngresII/"
II_WORK="/opt/Ingres/IngresII/"
II_DUMP="/opt/Ingres/IngresII/"
II_LOG_FILE="/opt/Ingres/IngresII/"
II_DUAL_LOG="/opt/Ingres/IngresII/"

# [ Ingres Configuration ]

II_INSTALLATION=II
II_CHARSET="UTF8"
II_TIMEZONE_NAME="NA-EASTERN"
II_TERMINAL="konsole"
II_DATE_FORMAT="US"
II_DATE_TYPE_ALIAS="ANSIDATE"
II_MONEY_FORMAT="L:$"
II_LOG_FILE_SIZE_MB="256"
II_ENABLE_SQL92="NO"
II_DEMO_DB_CREATE="YES"
II_START_ON_BOOT="YES"
II_USERID=
II_GROUPID=
II_UPGRADE_USER_DB="YES"
II_REMOVE_ALL_FILES="NO"
II_CONFIG_TYPE = "TXN"
```

Example Response File for Windows

The following example response file includes all parameter options and their default values for a Windows installation.

```
; Ingres Response File
; Generated by the Ingres Response File API
; Created: Thu Nov 08 13:14:24 2006
```

[Ingres Components]

```
II_COMPONENT_DBMS = "YES"
II_COMPONENT_NET = "YES"
II_COMPONENT_DOCUMENTATION = "YES"
II_COMPONENT_DOTNET = "YES"
II_COMPONENT_FRONTTOOLS = "YES"
II_COMPONENT_REPLICATOR = "NO"
II_COMPONENT_STAR = "NO"
II_COMPONENT_SPATIAL = "YES"
II_COMPONENT_JDBC_CLIENT = "YES"
```

[Ingres Locations]

```
II_SYSTEM = "C:\Program Files\Ingres\IngresII"
II_DATABASE = "C:\Program Files\Ingres\IngresII"
II_CHECKPOINT = "C:\Program Files\Ingres\IngresII"
II_JOURNAL = "C:\Program Files\Ingres\IngresII"
II_WORK = "C:\Program Files\Ingres\IngresII"
II_DUMP = "C:\Program Files\Ingres\IngresII"
II_LOG_FILE = "C:\Program Files\Ingres\IngresII"
II_DUAL_LOG = ""
II_LOCATION_DOTNET = "C:\Program Files\Ingres\Ingres .NET Data Provider\"
II_LOCATION_DOCUMENTATION = "C:\Program Files\Ingres\Ingres Documentation 9.3\"
```

[Ingres Configuration]

```
II_INSTALLATION = "II"
II_CHARSET = "WIN1252"
II_TIMEZONE_NAME = "NA-EASTERN"
II_TERMINAL = "IBMPCD"
II_DATE_FORMAT = "US"
II_DATE_TYPE_ALIAS = "ANSIDATE"
II_MONEY_FORMAT = "L:$"
II_LOG_FILE_SIZE_MB = "32"
II_ENABLE_SQL92 = "NO"
II_ADD_REMOVE_PROGRAMS = "YES"
II_ADD_TO_PATH = "YES"
II_INSTALL_ALL_ICONS = "YES"
II_START_INGRES_ON_COMPLETE = "YES"
II_START_IVM_ON_COMPLETE = "YES"
II_SERVICE_START_AUTO = "YES"
II_SERVICE_START_USER = ""
II_SERVICE_START_USERPASSWORD = ""
II_ENABLE_TCPIP = "YES"
II_ENABLE_NETBIOS = "NO"
II_ENABLE_SPX = "NO"
II_DEMO_DB_CREATE = "YES"
II_SETUP_ODBC_READONLY = "NO"
II_CONFIG_TYPE = "TXN"
```

[User Defined Properties]

Response File Parameters

Parameters in the response file determine how Ingres is to be installed and configured.

II_ADD_REMOVE_PROGRAMS

Valid on Windows

The II_ADD_REMOVE_PROGRAMS parameter defines whether Ingres is added to the Add/Remove Programs folder in Windows.

This parameter has the following options:

YES

(Default) Add Ingres to the Add/Remove Programs folder in Windows.

NO

Do not add Ingres to the Add/Remove Programs folder in Windows.

II_ADD_TO_PATH

Valid on Windows

The II_ADD_TO_PATH parameter defines whether to add Ingres to the Windows PATH.

This parameter has the following options:

YES

(Default) Add Ingres to the Windows PATH.

NO

Do not add Ingres to the Windows PATH.

II_CHARSET

The II_CHARSET parameter defines the Ingres character set (see page 26). This parameter cannot be changed without corrupting data.

Defaults:

Windows: WIN1252

UNIX and Linux: ISO88591

Note: In Linux, the character set value must be entered in uppercase.

II_CHECKPOINT

The II_CHECKPOINT parameter defines the location of the checkpoint files that serve as a static backup of the database. This parameter cannot be changed, even during installation updates. Specific databases can designate alternate locations for checkpoints as a parameter to the createdb command.

Default: Same as II_SYSTEM

II_COMPONENT_CORE

Valid on Windows

The II_COMPONENT_CORE parameter defines whether the core package for Linux is included in the Ingres installation. The core package is required by all other optional packages and must always be included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the Linux core package.

NO

Do not include the Linux core package.

II_COMPONENT_DBMS

Valid on Windows

The II_COMPONENT_DBMS parameter defines whether the Ingres DBMS Server package is included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the Ingres DBMS Server package.

NO

Do not include the Ingres DBMS Server package.

II_COMPONENT_DOCUMENTATION

Valid on Windows

The II_COMPONENT_DOCUMENTATION parameter defines whether the Ingres documentation package is included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the Ingres documentation package.

NO

Do not include the Ingres documentation package.

II_COMPONENT_DOTNET

Valid on Windows

The II_COMPONENT_DOTNET parameter defines whether the Ingres .NET Data Provider package is included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the Ingres .NET Data Provider package.

NO

Do not include the Ingres .NET Data Provider package.

II_COMPONENT_FRONTTOOLS

Valid on Windows

The II_COMPONENT_FRONTTOOLS parameter defines whether the Front Tools package is included in the Ingres installation. This package contains the following components:

- Ingres Embedded SQL/C
- Ingres Embedded SQL/COBOL
- Ingres Embedded SQL/FORTRAN
- Ingres Querying and Reporting Tools
- Ingres Visual DBA

This parameter has the following options:

YES

(Default) Include the Front Tools package.

NO

Do not include the Front Tools package.

II_COMPONENT_JDBC_CLIENT

Valid on Windows

The II_COMPONENT_JDBC_CLIENT parameter defines whether the JDBC driver is included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the JDBC driver.

NO

Do not include the JDBC driver.

II_COMPONENT_NET

Valid on Windows

The II_COMPONENT_NET parameter defines whether the Ingres Net package is included in the Ingres installation. This package includes the following items:

- Name Server
- Communications Server
- Data Access Server
- ODBC and JDBC drivers
- Terminal Monitors

This parameter has the following options:

YES

(Default) Include the Ingres Net package.

NO

Do not include the Ingres Net package.

II_COMPONENT_ODBC

Valid on Windows

The II_COMPONENT_ODBC parameter defines whether the ODBC Driver is included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the ODBC Driver.

NO

Do not include the ODBC Driver.

II_COMPONENT_REPLICATOR

Valid on Windows

The II_COMPONENT_REPLICATOR parameter defines whether the Ingres Replicator package is included in the Ingres installation.

This parameter has the following options:

YES

Include the Ingres Replicator package.

NO

(Default) Do not include the Ingres Replicator package.

II_COMPONENT_SPATIAL

Valid on Windows

The II_COMPONENT_SPATIAL parameter defines whether the Spatial Object Library, which allows spatial data types, is included in the Ingres installation.

This parameter has the following options:

YES

(Default) Include the Spatial Object Library.

NO

Do not include the Spatial Object Library.

II_COMPONENT_STAR

Valid on Windows

The II_COMPONENT_STAR parameter defines whether the Ingres Star package is included in the Ingres installation.

This parameter has the following options:

YES

Include the Ingres Star package.

NO

(Default) Do not include the Ingres Star package.

II_COMPONENT_VISION

Valid on Windows

The II_COMPONENT_VISION parameter defines whether Vision, a program for developing forms-based applications, is included in the Ingres installation.

This parameter has the following options:

YES

Include the Vision program.

NO

(Default) Do not include the Vision program.

II_CONFIG_TYPE

The II_CONFIG_TYPE parameter defines how Ingres is configured when installed.

This parameter has the following options:

TXN

(Default) Configures Ingres for a Transactional system (such as ERP or CRM systems, electronic banking, order processing, time clock systems, and e-commerce systems).

BI

Configures Ingres for a Business Intelligence system (reporting, analytics, data mining, business performance management, decision support, and predictive analytics).

CM

Configures Ingres for a Content Management system (storing, controlling, versioning, and publishing information including files, image media, audio, video, electronic documents, and Web content).

NULL

Configures Ingres for a high transaction isolation, low concurrency system (traditional Ingres defaults).

II_DATABASE

The II_DATABASE parameter defines the location of the Ingres master database (iibd) and the default location for database files.

Default: Same as II_SYSTEM

II_DATE_TYPE_ALIAS

The II_DATE_TYPE_ALIAS parameter controls whether the keyword DATE used to define the column data type refers to the INGRESDATE or the ANSIDATE data type.

This parameter has the following options:

ANSIDATE

Date columns use the ANSI date data type.

INGRESDATE

Date columns use the Ingres date data type.

Default: INGRESDATE if II_CONFIG_TYPE=NULL; otherwise ANSIDATE.

II_DATE_FORMAT

Valid on Windows, Linux

The II_DATE_FORMAT parameter defines the format used to display date values. The default US setting displays date *output* as dd-mm-yy. Legal *input* formats for US dates include:

- mm/dd/yy
- mm/dd/yyyy
- dd-mmm-yyyy
- mm-dd-yyyy
- yyyy.mm.dd
- yyyy_mm_dd
- mmddy
- mm-dd
- mm/dd

This parameter has the following options:

Setting	Output Format
US	dd-mm-yy
ISO	yymmdd
MULTINATIONAL	dd/mm/yy
SWEDEN/FINLAND	yyyy-mm-dd

Setting	Output Format
GERMAN	dd.mm.yy
DMY	dd-mmm-yyyy
MDY	mmmm-dd-yyyy
YMD	yyyy-mmm-dd

Default: US

II_DEMODB_CREATE

The II_DEMODB_CREATE parameter defines whether to install the Ingres demonstration program and create the demonstration database.

This parameter has the following options:

YES

(Default) Install the demonstration program and create the demonstration database.

NO

Do not install the demonstration program.

Note: The demonstration program requires both the DBMS and .NET Data Provider components. If only the DBMS is installed, the demonstration database (demodb) will be created, but the demonstration program will not be installed.

II_DESTROY_TXLOG

Valid on Windows

The II_DESTROY_TXLOG parameter defines whether the Ingres transaction log file is destroyed and recreated during an update of Ingres.

This parameter has the following options:

YES

Destroy and recreate the Ingres transaction log file during an update of Ingres.

NO

(Default) Do not destroy and recreate the Ingres transaction log file during an update of Ingres.

II_DUAL_LOG

The II_DUAL_LOG parameter defines the location of the backup (secondary) transaction log file. By default, dual transaction logging is disabled (parameter is set to blank ("")). If you desire dual logging, configure the II_SYSTEM location value for this parameter setting.

Default: ""

II_DUMP

The II_DUMP parameter defines the location of dump files used to store changes made during an online checkpoint.

Default: Same as II_SYSTEM

II_ENABLE_NETBIOS

Valid on Windows

The II_ENABLE_NETBIOS parameter defines whether the NETBIOS networking protocol is initialized when the Communications Server starts.

This parameter has the following options:

YES

Initialize the NETBIOS networking protocol when the Communications Server starts.

NO

(Default) Do not initialize the NETBIOS networking protocol when the Communications Server starts.

II_ENABLE_SPX

Valid on Windows

The II_ENABLE_SPX parameter defines whether the SPX networking protocol is initialized when the Communications Server starts.

This parameter has the following options:

YES

Initialize the SPX networking protocol when the Communications Server starts.

NO

(Default) Do not initialize the SPX networking protocol when the Communications Server starts.

II_ENABLE_SQL92

The II_ENABLE_SQL92 parameter defines whether the SQL92 standard setting is enabled.

This parameter has the following options:

YES

Enable the SQL92 standard.

NO

(Default) Do not enable the SQL92 standard.

II_ENABLE_TCPIP

Valid on Windows

The II_ENABLE_TCPIP parameter defines whether the TCP_IP networking protocol is initialized when the Communications Server starts.

This parameter has the following options:

YES

(Default) Initialize the TCP_IP networking protocol when the Communications Server starts.

NO

Do not initialize the TCP_IP networking protocol when the Communications Server starts.

II_GROUPID

Valid on UNIX, Linux

The II_GROUPID parameter specifies the group ID to which the current user's user ID belongs and that owns the Ingres installation. The ID is added to the system if it does not exist.

This parameter has the following option:

group id

Default: Current user's group ID

II_INSTALL_ALL_ICONS

Valid on Windows

The II_INSTALL_ALL_ICONS parameter defines whether to create the Ingres program group and associated icons during the Ingres installation.

This parameter has the following options:

YES

(Default) Create the Ingres program group and its associated icons.

NO

Do not create the Ingres program group and its associated icons.

II_INSTALLATION

The II_INSTALLATION parameter defines the Ingres installation code. The installation code is a two-character code that identifies a specific installation on a node, and allows all Ingres processes and images to be installed and shared successfully.

If you have more than one installation on the same node, each installation on that node must have a unique installation code. For example, you can install and run a new version of Ingres under one installation code, while maintaining an existing older installation under a different installation code on the same computer or network node.

This parameter has the following options:

[A-Z], [A-Z] [0-9]

The first character of the installation code must be a letter; the second character can be a letter or numeral.

Default: II

II_JOURNAL

The II_JOURNAL parameter defines the location of the journal files, which provide a dynamic record of changes made to Ingres databases since the last checkpoint. This parameter cannot be changed, even during installation updates. Specific databases can designate alternate locations for journals as a parameter to the createdb command.

Default: Same as II_SYSTEM

II_LOCATION_DOCUMENTATION

Valid on Windows

The II_LOCATION_DOCUMENTATION parameter defines the location of the Ingres documentation set.

Default: C:\Program Files\Ingres\Ingres Documentation *version*

II_LOCATION_DOTNET

Valid on Windows

The II_LOCATION_DOTNET parameter defines the location of the Ingres .NET Data Provider files.

Default: C:\Program Files\Ingres\Ingres .NET Data Provider

II_LOG_FILE

The II_LOG_FILE parameter defines the location of the transaction log file.

Default: Same as II_SYSTEM

II_LOG_FILE_SIZE_MB

The II_LOG_FILE_SIZE_MB parameter defines the size of the Ingres primary transaction log in MB.

This parameter has the following options:

16-2048

Specifies a range of 16 to 2048 MB for the primary transaction log.

Default: 256

II_MONEY_FORMAT

Valid on Windows, Linux

The II_MONEY_FORMAT parameter defines the format of monetary outputs. The symbol to the left of the colon indicates the location of the currency symbol. It must be "L" for a leading currency symbol or a "T" for a trailing currency symbol. The symbol to the right of the colon is the currency symbol you want displayed. Currency symbols can contain up to 4 physical characters.

For example:

Logical Definition	Result
L:\$	\$100
T:DM	100DM
T:F	100F

Default: L:\$

II_REMOVE_ALL_FILES

Valid on Linux

The II_REMOVE_ALL_FILES parameter defines whether all Ingres files are removed during an un-install of Ingres.

This parameter has the following options:

YES

Remove all Ingres files during an un-install of Ingres.

NO

(Default) Do not remove all Ingres files during an un-install of Ingres.

II_SERVICE_START_AUTO

Valid on Windows

The II_SERVICE_START_AUTO parameter defines whether the Ingres Service is started automatically when the machine on which Ingres is installed is started.

This parameter has the following options:

YES

(Default) Start the Ingres Service automatically when the machine on which Ingres is installed is started.

NO

Do not start the Ingres Service automatically when the machine on which Ingres is installed is started.

II_SERVICE_START_USER

Valid on Windows

The II_SERVICE_START_USER parameter defines the user who can start Ingres as a service.

This parameter has the following option:

user

Default: Current user

II_SERVICE_START_USERPASSWORD

Valid on Windows

The II_SERVICE_START_USERPASSWORD parameter defines the password required to start Ingres as a service.

This parameter has the following option:

password

Default: Current user's password

II_START_INGRES_ON_COMPLETE

Valid on Windows

The II_START_INGRES_ON_COMPLETE parameter defines whether to start Ingres after the embedded Ingres installation completes.

This parameter has the following options:

YES

(Default) Start Ingres after the embedded Ingres installation completes.

NO

Do not start Ingres after the embedded Ingres installation completes.

II_START_IVM_ON_COMPLETE

Valid on Windows

The II_START_IVM_ON_COMPLETE parameter defines whether to start the Ingres Visual Manager utility after the embedded Ingres installation completes.

This parameter has the following options:

YES

(Default) Start Ingres Visual Manager after the embedded Ingres installation completes.

NO

Do not start Ingres Visual Manager after the embedded Ingres installation completes.

II_START_ON_BOOT

Valid on Linux

The II_START_ON_BOOT parameter defines whether Ingres starts when the Linux server starts up.

This parameter has the following options:

YES

(Default) Start Ingres when the Linux server starts up.

NO

Do not start Ingres when the Linux server starts up.

II_SYSTEM

Valid on Windows, UNIX

The II_SYSTEM parameter defines the location of the Ingres system (executable) files.

Defaults:

Windows: C:\Program Files\Ingres\IngresII\

Linux: /opt/Ingres/IngresII/

II_TERMINAL

Valid on Windows, Linux (for RPM only)

The II_TERMINAL parameter defines the terminal type.

This parameter has the following options:

Platform	Options
ANSI	ansif ansinf iris-ansi
BULL	bull12 bull24 bullbds1 bullvtu10 bullwv

Platform	Options	
DG	dg100em dg220em dgxterm	
HP	hp2392 hp70092 hpterm	
ICL	icl12 icl34 icl5 icldrs	
Linux	konsole konsolel	
MWS	mws00 mws01 mws02 mws03	mws04 mws05 mws06
PC	ibm5151f ibmpc ibmpcd pc-220	pc-305 pc-386 pckermi winpcalt
SUN	dtterm gnome-sun4 gnome-sun suncmdf sunf	sunk sunm suntime5 xsun
VT	vt100f vt100hp vt100i vt220nk	vt200i vt220ak vt220
Other	io8256 h19f h19nk wview wy60at xmlname 97801f at386	pt35 pt35t tk4105 m30n m91e mac2 cbf frs

Defaults:

Windows: IBMPCD

Linux: konsolel

II_TIMEZONE_NAME

The II_TIMEZONE_NAME parameter defines the operating system time zone (see page 24) for the Ingres installation. This parameter tells Ingres what adjustments to make for Daylight Savings Time.

Defaults:

Windows: NA-EASTERN

UNIX: NA-PACIFIC

II_UPGRADE_USER_DB

The II_UPGRADE_USER_DB parameter defines whether the existing user databases are upgraded automatically.

This parameter has the following options:

YES

(Default) Upgrade the existing user databases automatically.

NO

Do not upgrade the existing user databases automatically.

II_USERID

Valid on UNIX, Linux

The II_USERID parameter defines the user ID of the user that owns the Ingres installation. The ID is added to the system if it does not exist.

This parameter has the following option:

user id

Default: User ID of the user who installs Ingres

II_WORK

The II_WORK parameter defines the location of temporary work files created during external sorts, table modifies and other DBMS Server operations.

Default: Same as II_SYSTEM

Index

.

.NET Data Provider • 41

A

access

the instance • 46

to database and tools • 79, 93, 94

using shadow passwords • 79

accessdb command • 79, 93

automatic startup • 78, 93

B

Bridge Server • 31

C

character set • 26

checkpoint

files • 29

client installation • 24

client-only disk configuration • 24

cluster

VMS • 89

Communications Server • 31, 61

computer names • 32

configuration type • 47

Configuration-By-Forms • 55

configurations, major • 61

customization after installation • 55

D

Data Access Server • 31

database

files • 29

date_alias • 28

DBMS Server

installation parameters • 29

networked • 61

disk configurations • 23, 24

documentation • 41

domain • 10

dump files • 29

E

environment variables

II_CHECKPOINT • 29

II_DATABASE • 29

II_DUMP • 29

II_JOURNAL • 29

II_LOG_FILE_SIZE_MB • 29

II_SYSTEM • 28, 65, 74, 91

II_TIMEZONE_NAME • 24

II_USERID • 28

II_WORK • 29

F

file locations • 75

H

High Availability Option • 101, 105, 115

history log file • 87

I

II_CHARSET • 29

II_CHECKPOINT • 29

II_CONFIG_TYPE • 47

II_DATABASE • 29

II_DISTRIBUTION • 66

II_DUMP • 29

II_JOURNAL • 29

II_LOG_FILE • 29

II_SYSTEM • 28, 65, 74, 88, 91

II_TIMEZONE_NAME • 24, 29

II_USERID • 28

II_WORK • 29

ingbuild command • 67

ingmknfs command • 72

ingsetenv command • 55

ingstart and ingstop commands • 53

installation

code • 28

history log file • 95

owner • 10, 65

parameters, general • 28

installing Ingres

-
- on UNIX • 65
 - on Windows • 41

J

- journal files • 29

L

- Linux
 - installing Ingres on • 33
- location • 75
- log file
 - as a raw device • 75
- logging system
 - making changes to • 55
- logicals
 - concealed • 88, 89

M

- multiple instances • 10, 87

N

- Name Server • 31
- names • 32
- netutil utility • 55
- network protocols • 31
- NFS client • 24, 72, 74

O

- OpenVMS
 - hardware requirements • 125
 - process resources • 129
 - quotas • 130
 - required privileges • 131
 - SYSGEN parameters • 126
 - system requirements • 125

P

- parameters
 - DBMS Server installation • 29
 - general installation • 28
- password
 - DBMS Server • 55
 - shadow • 79

Q

- quick start • 55

R

- raw log file • 75, 76
- Recovery Server • 31
- response file • 35, 36, 37, 38, 42, 67, 133, 136
- RPM • 33, 34

S

- servers
 - Ingres • 31
 - Service Manager (Windows) • 52
 - setenv command • 63, 66
 - setup program • 91
 - shadow passwords • 79
 - Star Server • 31
 - start Ingres • 49, 50, 51, 52, 53
 - startup, automatic • 78, 93
 - SYSGEN parameters • 90, 126
 - system administrator account • 10
 - create • 60
 - owner • 28
 - system files • 28

T

- TERM_INGRES • 63, 85
- terminal
 - define • 63, 85
- time zone • 24
- transaction log file
 - as a raw device • 75

U

- UIC • 90
- uninstall Ingres • 57, 58
- upgrading • 65, 91
- user
 - access to tools • 79, 94
 - ID • 28
 - login file • 79, 80
 - names • 32

V

Vision • 41

Visual Manager • 51

W

work files • 29