

Ingres® 2006 Release 2

Installation Guide

INGRES®

February 2007

This documentation and related computer software program (hereinafter referred to as the "Documentation") is for the end user's informational purposes only and is subject to change or withdrawal by Ingres Corporation ("Ingres") at any time.

This Documentation may not be copied, transferred, reproduced, disclosed or duplicated, in whole or in part, without the prior written consent of Ingres. This Documentation is proprietary information of Ingres and protected by the copyright laws of the United States and international treaties.

Notwithstanding the foregoing, licensed users may print a reasonable number of copies of this Documentation for their own internal use, provided that all Ingres copyright notices and legends are affixed to each reproduced copy. Only authorized employees, consultants, or agents of the user who are bound by the confidentiality provisions of the license for the software are permitted to have access to such copies.

This right to print copies is limited to the period during which the license for the product remains in full force and effect. The user consents to Ingres obtaining injunctive relief precluding any unauthorized use of the Documentation. Should the license terminate for any reason, it shall be the user's responsibility to return to Ingres the reproduced copies or to certify to Ingres that same have been destroyed.

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 USE 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 use of any product referenced in this Documentation and this Documentation is governed by the end user's applicable license agreement.

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 © 2007 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	7
An Ingres Instance.....	7
Instance ID	7
System Administrator Account	8
Ingres Files	9
File Location Guidelines.....	10
DBMS Server Disk Configurations	11
Four-Disk DBMS Server Configuration.....	12
Three-Disk DBMS Server Configuration	12
Two-Disk DBMS Server Configuration	12
One-Disk DBMS Server Configuration	13
Client-only Disk Configuration	13
NFS Client Disk Configuration (UNIX)	13
Installation Parameters	13
General Installation Parameters	13
DBMS Server Installation Parameters	14
World Region and Time Zone	15
Time Zone Names.....	15
Supported Character Sets	17
Ingres Servers.....	19
Valid Computer, Directory, and User Names.....	20
Chapter 2: Installing Ingres	21
How You Install Ingres.....	21
Installing Ingres for Linux	21
Install Ingres for Linux Using the Installation Wizard.....	22
Installing Ingres for Linux at the Command Line	22
Install Ingres for Windows.....	29
Separately Installed Components.....	29
Advanced Component Selection	29
Perform a Silent Install on Windows	30
Install Ingres for UNIX.....	32
Install Ingres for OpenVMS.....	33
How You Access the Instance on Linux and UNIX	34

Chapter 3: Starting Ingres	35
Start the Ingres Installation on Linux and UNIX	35
Startup and Shutdown on Windows	36
Start and Stop Ingres with Ingres Visual Manager	37
Start and Stop Ingres with the Ingres Service Manager	38
Start the Ingres Installation on OpenVMS	39
Getting Started with Databases and Applications	39
Chapter 4: Understanding Post-installation Tasks	41
How You Further Customize Your Instance	41
How You Prepare Your Installation for General Use	42
How You Safely Uninstall Ingres	43
Uninstall Ingres for Linux Using the Installation Wizard	43
uninstall_inger Command—Uninstall Ingres for Linux	44
Uninstall Ingres for Windows	44
Appendix A: Installing Ingres on UNIX	45
How You Prepare for Installation on UNIX	45
How You Create the System Administrator Account on UNIX	46
Major Configuration Options	46
How You Install and Configure Ingres in a UNIX Environment	48
Installation Modes	48
inger_express_install Command—Install Ingres Quickly	49
Install Ingres in Interactive Mode	49
Install Ingres in Command Line Mode	50
Ingbuild Command—Install Ingres	50
Run the Setup Programs	53
How You Install and Configure NFS Clients	54
Set Up Each NFS Client Individually	54
Set Up NFS Clients from the DBMS Server	56
Set TERM_INGRES	57
Set II_DISTRIBUTION	58
Post-Installation Tasks	58
Transaction Log File as a Raw Device	58
Automatic Startup Command	61
Allow Access on Systems Using Shadow Passwords	62
How You Establish User Access to Tools and Databases	62
View Contents of Your Current Installation	64

Appendix B: Installing Ingres on OpenVMS	67
How You Install and Configure Ingres (First Time Installers).....	67
How You Install and Configure Ingres (Experienced Installers).....	68
Define the TERM_INGRES Logical	69
Major Configuration Options	69
Multiple Instances on One Node	71
File Locations and Logicals.....	72
Define II_SYSTEM as a Logical.....	72
Define a Concealed, Rooted Logical	73
Guidelines for VMS Cluster Configuration.....	73
Owner of the System Administrator Account	74
UIC for an Installation	74
Run the Setup Programs	75
Post-Installation Tasks.....	76
How You Add an Automatic Startup Command	77
How You Establish User Access to Tools and Databases	77
Installation History Log File	78
Ingres Cluster Solution for OpenVMS	80
Requirements for the Ingres Cluster Solution on OpenVMS	80
How You Prepare to Install the Ingres Cluster Solution for OpenVMS	81
How You Install the Ingres Cluster Solution for OpenVMS	81
How Client Applications Access an Ingres Cluster	82
Class Node Affinity (CNA)	83
Appendix C: Configuring Ingres Cluster Solution for Linux	85
Ingres Cluster Solution Overview.....	86
Requirements for the Ingres Cluster Solution	87
Preparation for Installing the Ingres Cluster Solution	88
How You Install the Ingres Cluster Solution for Linux.....	88
How Client Applications Access an Ingres Cluster	90
Class Node Affinity (CNA)	90
Global File System and Database Creation	91
Appendix D: System Requirements for OpenVMS	93
System Hardware Requirements on OpenVMS	93
SYSGEN Parameters that Affect Operation	94
Recommended OpenVMS Process Resources	98
OpenVMS Quotas and Ingres Parameters	99
Required OpenVMS Privileges.....	100

Chapter 1: Understanding Installation Considerations

This section contains the following topics:

- [An Ingres Instance](#) (see page 7)
- [System Administrator Account](#) (see page 8)
- [Ingres Files](#) (see page 9)
- [File Location Guidelines](#) (see page 10)
- [DBMS Server Disk Configurations](#) (see page 11)
- [Installation Parameters](#) (see page 13)
- [World Region and Time Zone](#) (see page 15)
- [Supported Character Sets](#) (see page 17)
- [Ingres Servers](#) (see page 19)
- [Valid Computer, Directory, and User Names](#) (see page 20)

An Ingres Instance

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 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.

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.

File Location Guidelines

Before installing Ingres, you must decide on locations for the Ingres files. Choose these locations carefully because they cannot be easily changed once specified (except for the location of the transaction log).

During installation, each location is stored in an Ingres environment variable/logical (for example, the location for the system files is stored in `II_SYSTEM`).

Use the following guidelines when choosing file locations:

- **System files (II_SYSTEM)**

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.

- **Database files (II_DATABASE)**

The location for your databases contains the master database (`iidbdb`), 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 this 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 optimizing disk I/O.
- On systems with three or more disks, do not place the database files on the same disk as your operating system because it forces database scans to compete with system operations for I/O.
- Do not place database files on a device that is likely to become full as databases are added. Full disks can become fragmented, causing slow disk retrievals and degraded system performance.

- **Checkpoint, journal, and dump files (II_CHECKPOINT, II_JOURNAL, II_DUMP)**

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 device on the same disk as the database files. Storing data and backups on the same device provides poor insurance against disk failure. On single-disk systems, we recommend checkpointing to magnetic tape.

- **Temporary work files (II_WORK)**

Work files can reside on the same device as the checkpoint files because work files are useful in recovery only if the associated checkpoint is also available. For highest performance, however, assign your temporary work files to a different physical device than your database, checkpoint, journal, or dump files.

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

- **Transaction and backup transaction logs**

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

- If possible, avoid installing the primary and backup transaction log files on I/O-bound disks, because this can increase data acquisition times of the Recovery Server and archiver process and slow down all users.
- If possible, put the transaction log file on separate disks from your database, checkpoint, dump, and journal files so that you can recover your unsaved, committed transactions if any of these disks fail.

Locations for these files are defined by the log_file (primary) and dual_log (backup) configuration parameters. You can add, modify, or delete locations for your primary and dual transaction log files using the Configuration-By-Forms utility.

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).

Warning! We strongly recommend that you do not use ReiserFS for database locations. Severe performance degradation was noticed when using this file system.

Note: The II_SYSTEM location should not be placed on an OCFS-1 CFS.

DBMS Server Disk Configurations

Typical disk configurations for an Ingres installation are described here.

For any DBMS Server configuration, you can set up an optional backup transaction log device on a separate disk from the primary transaction log. This setup enables recovery of unsaved, committed transactions if the primary transaction log device fails.

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—Checkpoint, journal, and dump files
- Disk 3—Product system files, and transaction log work files
- Disk 4—Databases

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, checkpoint, journal, and dump files
- Disk 2—Product system files, transaction log, and work files
- Disk 3—Databases

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

If Disk 3 fails, you can recover databases and committed transactions. If Disk 2 fails, you can recover your committed transactions if you have a backup transaction log; if you have no backup log, you will lose committed transactions that were not written to the database.

Two-Disk DBMS Server Configuration

A two-disk system is the minimal recommended configuration because it avoids a single point of failure:

- Disk 1—All Ingres files except for those on Disk 2
- Disk 2—Databases, optional backup transaction log file

If Disk 2 fails, you can recover databases and committed transactions. However, if Disk 1 fails and you do not have a secondary log device, you will lose unsaved, committed transactions.

One-Disk DBMS Server Configuration

A single-disk system is a high-risk setup and not recommended. If the disk fails, you could lose all your data. On single-disk systems, you should checkpoint to magnetic tape.

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. The Ingres DBMS Server executables reside on the networked DBMS Server instance.

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 `imgmknfs` 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.

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:

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:

Database location (II_DATABASE)

Defines the location for the Ingres master database (iadbdb) 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 (32768 KB 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.

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	North America	Southeast Asia
GMT	NA-PACIFIC	INDONESIA-WEST
GMT1	NA-MOUNTAIN	INDONESIA-CENTRAL
GMT2	NA-CENTRAL	INDONESIA-EAST
GMT3	NA-EASTERN	MALAYSIA
GMT4	NA-ALASKA	PHILIPPINES
Asia	Canada-Atlantic	SINGAPORE
INDIA	CANADA-NEWFOUNDLAND	THAILAND
JAPAN	CANADA-YUKON	VIETNAM
KOREA	MEXICO-GENERAL	GMT7
HONG-KONG	MEXICO-BAJANORTE	GMT8
PAKISTAN	MEXICO-BAJASUR	GMT9
PRC	North-Atlantic	GMT-Offset
ROC	EUROPE-WESTERN	GMT-12
GMT5	EUROPE-CENTRAL	GMT-11
GMT6	EUROPE-EASTERN	GMT-10
GMT7	IRELAND	GMT-9
GMT8	MOSCOW	GMT-9-and-half
GMT9	POLAND	GMT-8
GMT10	TURKEY	GMT-7
GMT11	UNITED-KINGDOM	GMT-6
Australia	GMT	GMT-5
AUSTRALIA-LHI	GMT1	GMT-5-and-half
AUSTRALIA-NORTH	GMT2	GMT-4
AUSTRALIA-WEST	GMT3	GMT-3
AUSTRALIA-SOUTH	South America	GMT-3-and-half
AUSTRALIA-TASMANIA	BRAZIL-EAST	GMT-2
AUSTRALIA-QUEENSLAND	BRAZIL-WEST	GMT-2-and-half
AUSTRALIA-VICTORIA	BRAZIL-ACRE	GMT-1
AUSTRALIA-NSW	BRAZIL-DENORONHA	GMT
AUSTRALIA-YANCO	CHILE-CONTINENTAL	GMT1
Middle East	CHILE-EASTER-ISLAND	GMT2
EGYPT	GMT6	GMT3
IRAN	GMT5	GMT3-and-half
ISRAEL	GMT4	GMT4
KUWAIT	GMT3	GMT5
SAUDI-ARABIA	South Pacific	GMT5-and-half
GMT2	NEW-ZEALAND	GMT6
GMT3	US-HAWAII	GMT7
GMT4	GMT10	GMT8
	GMT11	GMT9
	GMT12	GMT9
	GMT-12	GMT10
	GMT-11	GMT10
	GMT-10	GMT11
		GMT12
		GMT13

Supported Character Sets

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 for the non-Unicode character data types are as follows:

Character Set	Description	Format
ALT	Support of Cyrillic on DOS	Single byte
ARABIC	Arabic-449-Plus	Single byte
CHINESET	Traditional Chinese - Taiwan	Double byte
CHINESES	Simplified Chinese - PRC	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

Character Set	Description	Format
IS885915	ISO 8859/2 (Latin and some Greek). Identical to ISO 8859/1 Latin, except for eight characters, including the Euro currency symbol (€, Unicode U+20AC).	Single byte
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
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
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
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
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.

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.

ICE Server

Allows developers to build World Wide Web applications that can access enterprise-wide corporate data, through the Ingres Web Deployment Option.

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 Distributed Option.

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 21)
- [Installing Ingres for Linux](#) (see page 21)
- [Install Ingres for Windows](#) (see page 29)
- [Install Ingres for UNIX](#) (see page 32)
- [Install Ingres for OpenVMS](#) (see page 33)
- [How You Access the Instance on Linux and UNIX](#) (see page 34)

How You Install Ingres

If you are installing Ingres for the first time, you should be familiar with the Installation Considerations (see page 7) 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 10) and DBMS Server Disk Configurations (see page 11).
3. Install the software.
4. Perform post-installation tasks (see page 41).

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.

Install Ingres for Linux Using the Installation Wizard

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

To start the 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 [instance_ID] [instance_location]
```

instance_ID

Specifies the Ingres instance ID to use for the instance.

instance_location

Identifies the directory into which you want to install Ingres.

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, but with an instance ID of A1:

```
ingres_express_install A1
```

Note: If you set `II_RESPONSE_FILE` to point to a valid response file (see page 25), 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 Custom Install

The response file defines the configuration for your Ingres instance. You can use any name for the response file.

Each entry in the response file must be on a separate line and is a name-value pair, as follows:

variable=*value*

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

Commonly used variables are listed in Installation Parameters (see page 13).

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=KANJIEUC
```

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 24).

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, `ca-ingres` (for Ingres r3) or `ingres2006` (for Ingres 2006).

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 --replacefile --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, 3.0.3 for Ingres r3, or 9.0.4 for Ingres 2006).

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 in response file mode 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 install process will fail.

4. Click Finish to create the response file.

If you want to review the response file parameter settings in a text editor, click Yes when prompted. Otherwise, click No to exit the Ingres Setup Wizard.



Once created, a response file can be opened with any text editor, reviewed, and modified manually if changes are required to any of the values selected during the response file creation process.

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 45)

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 67)

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.

Chapter 3: Starting Ingres

This section contains the following topics:

[Start the Ingres Installation on Linux and UNIX](#) (see page 35)

[Startup and Shutdown on Windows](#) (see page 36)

[Start the Ingres Installation on OpenVMS](#) (see page 39)

[Getting Started with Databases and Applications](#) (see page 39)

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 62) 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

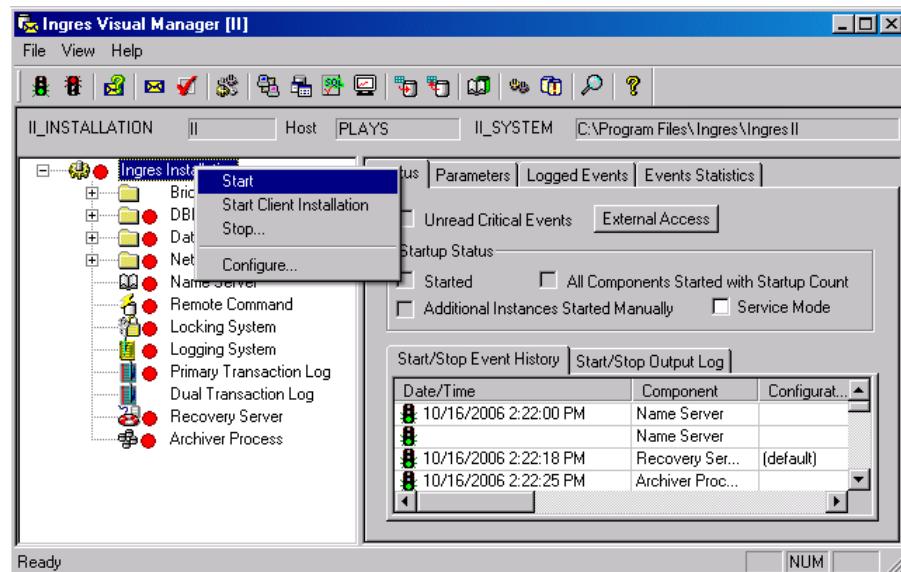
Note: Due to tightened security on the Microsoft Windows Vista platforms, we must impose the following restriction when using Ingres on Windows Vista: All Ingres tools and applications must be started from the Ingres Command Prompt, which is located in the Ingres Program Group under the Vista Start menu. 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.

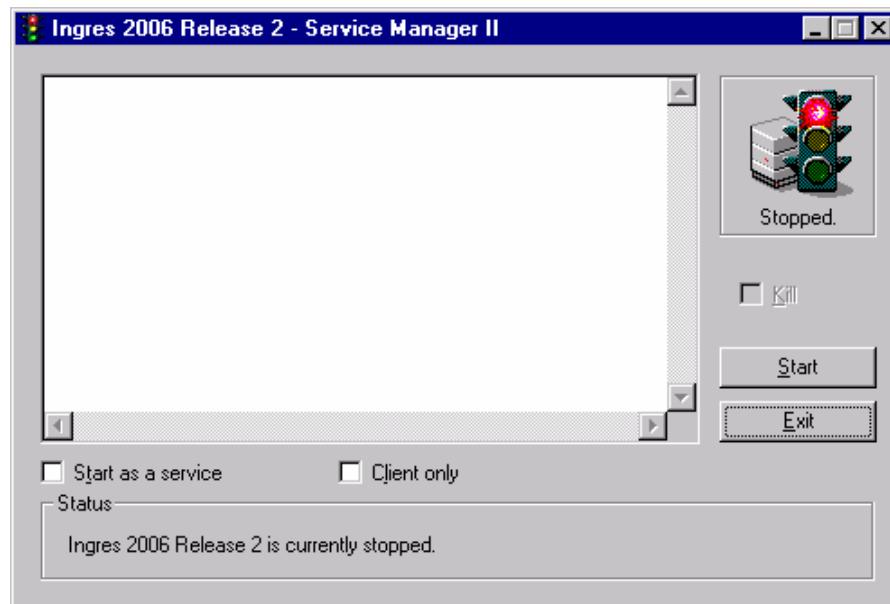
Start and Stop Ingres with the Ingres Service Manager

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. Check the Start as a service check box. Check 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 \(see page 41\)](#)
- [How You Prepare Your Installation for General Use \(see page 42\)](#)
- [How You Safely Uninstall Ingres \(see page 43\)](#)

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 58) 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 (ifdbdb) if it becomes corrupted, make sure that ifdbdb 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 chapter "Creating a Database."

Note: You do not need to create the ifdbdb 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 \(UNIX\) \(see page 58\)](#)

[Post-Installation Tasks \(VMS\) \(see page 76\)](#)

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 packages" removes only those files that are distributed with Ingres; it does not remove database and configuration files. To remove all files, check the "Remove all files" option.

uninstall_inger Command—Uninstall Ingres for Linux

The `uninstall_inger` 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_inger {[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/inger` 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 45)
- [How You Install and Configure Ingres in a UNIX Environment](#) (see page 48)
- [Installation Modes](#) (see page 48)
- [ingres express install Command—Install Ingres Quickly](#) (see page 49)
- [Install Ingres in Interactive Mode](#) (see page 49)
- [Install Ingres in Command Line Mode](#) (see page 50)
- [Run the Setup Programs](#) (see page 53)
- [How You Install and Configure NFS Clients](#) (see page 54)
- [Set TERM_INGRES](#) (see page 57)
- [Set II DISTRIBUTION](#) (see page 58)
- [Post-Installation Tasks](#) (see page 58)

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 `imgmknfs` 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. **(Optional) Set environment variables**—Define default values for your terminal type and distribution medium before running the install program, as described in Set TERM_INGRES (see page 57) and Set II_DISTRIBUTION (see page 58).
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 58).
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.

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.

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.

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 Setup can then use several of the values you specified for the DBMS Server.

To run the setup programs from the Currently Installed Components screen

1. Choose SetupAll to run the setup programs for all components that require it.

The setup programs execute one after another, in an appropriate order.

Note: If you need to set up components individually, highlight the appropriate row for the component you want to set up and choose Setup from the menu.

2. Respond to the program prompts.

When setting up required parameters, default values sometimes appear in brackets after the prompt. To accept a default value, press Return. Otherwise, enter an appropriate value and press Return.

The Setup program(s) prepare each component for use.

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:

```
% imgmknfs
```

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:

```
% imgmknfs 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:

```
% imgmknfs 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*).

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
```

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
```

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 startup 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 *System Administrator 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
```

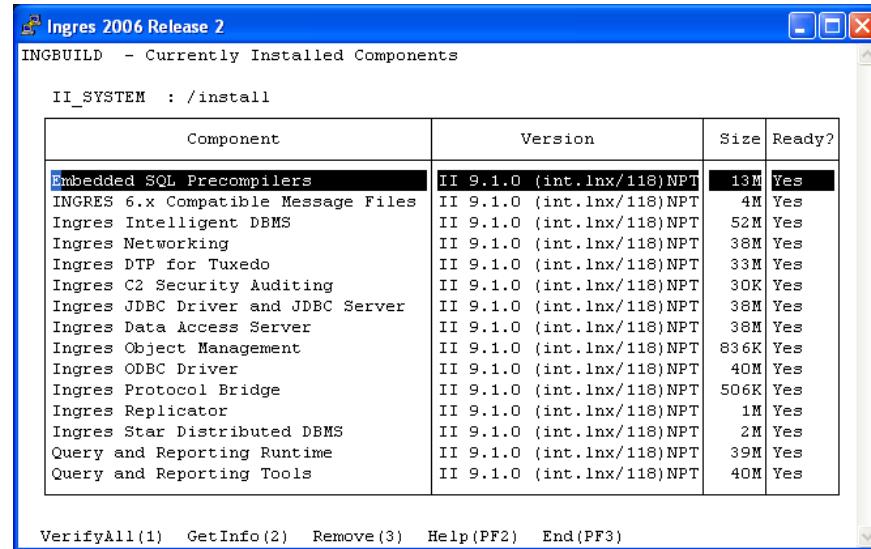
The initial screen of the install utility appears, with the following menu options:

```
CustomInstall PackageInstall Current Help Quit
```

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

4. Choose Current.

The Currently Installed Components screen appears.



The screenshot shows a window titled "Ingres 2006 Release 2" with the sub-title "INGBUILD - Currently Installed Components". The window displays a table of installed components. The table has columns for Component, Version, Size, and Ready?. The "Ready?" column contains "Yes" for all entries. The components listed are: Embedded SQL Precompilers, INGRES 6.x Compatible Message Files, Ingres Intelligent DBMS, Ingres Networking, Ingres DTP for Tuxedo, Ingres C2 Security Auditing, Ingres JDBC Driver and JDBC Server, Ingres Data Access Server, Ingres Object Management, Ingres ODBC Driver, Ingres Protocol Bridge, Ingres Replicator, Ingres Star Distributed DBMS, Query and Reporting Runtime, and Query and Reporting Tools. The "Size" column shows values such as 13M, 4M, 52M, 38M, 33M, 30K, 38M, 38M, 836K, 40M, 506K, 1M, 2M, 39M, and 40M. The "Version" column shows values such as II 9.1.0 (int.lnx/118)NPT. At the bottom of the window, there are buttons for VerifyAll(1), GetInfo(2), Remove(3), Help(PF2), and End(PF3).

Component	Version	Size	Ready?
Embedded SQL Precompilers	II 9.1.0 (int.lnx/118)NPT	13M	Yes
INGRES 6.x Compatible Message Files	II 9.1.0 (int.lnx/118)NPT	4M	Yes
Ingres Intelligent DBMS	II 9.1.0 (int.lnx/118)NPT	52M	Yes
Ingres Networking	II 9.1.0 (int.lnx/118)NPT	38M	Yes
Ingres DTP for Tuxedo	II 9.1.0 (int.lnx/118)NPT	33M	Yes
Ingres C2 Security Auditing	II 9.1.0 (int.lnx/118)NPT	30K	Yes
Ingres JDBC Driver and JDBC Server	II 9.1.0 (int.lnx/118)NPT	38M	Yes
Ingres Data Access Server	II 9.1.0 (int.lnx/118)NPT	38M	Yes
Ingres Object Management	II 9.1.0 (int.lnx/118)NPT	836K	Yes
Ingres ODBC Driver	II 9.1.0 (int.lnx/118)NPT	40M	Yes
Ingres Protocol Bridge	II 9.1.0 (int.lnx/118)NPT	506K	Yes
Ingres Replicator	II 9.1.0 (int.lnx/118)NPT	1M	Yes
Ingres Star Distributed DBMS	II 9.1.0 (int.lnx/118)NPT	2M	Yes
Query and Reporting Runtime	II 9.1.0 (int.lnx/118)NPT	39M	Yes
Query and Reporting Tools	II 9.1.0 (int.lnx/118)NPT	40M	Yes

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 67)
- [How You Install and Configure Ingres \(Experienced Installers\)](#) (see page 68)
- [Define the TERM_INGRES Logical](#) (see page 69)
- [Major Configuration Options](#) (see page 69)
- [Multiple Instances on One Node](#) (see page 71)
- [File Locations and Logicals](#) (see page 72)
- [Owner of the System Administrator Account](#) (see page 74)
- [Run the Setup Programs](#) (see page 75)
- [Post-Installation Tasks](#) (see page 76)
- [Ingres Cluster Solution for OpenVMS](#) (see page 80)

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 69).
2. Install the software as described in Install Ingres for OpenVMS (see page 33).

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 75) (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 39).

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 72) 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 33).

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 (see page 80).

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 Distributed Option 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 Distributed Option 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:[sysexe]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, except when accessing a database through a server class using "class node affinity"
- Update mode locks, except when accessing a database through a server class using "class node affinity"
- 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 and a nickname.

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 nickname is an optional simple alias, which you can use in any context in which you could specify a nodename parameter. The nickname appears in the error log instead of the machine name.

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 and nickname. 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.

Class Node Affinity (CNA)

Ingres allows the creation of server classes that function as regular DBMS servers but can be configured for specialized situations. The server parameter `class_node_affinity`, if set for a server class, allows servers in this class to be started on only one node at a time.

The configuration name and server class name for the default CNA classes generated is `cnnn`, where *nn* is the node number, zero padded.

While `iimkcluster` and `iisunode` set up a separate CNA class for each node they are run on, these classes are not bound to the node they were defined on, but can be started on any node. In addition, any database (except `iidbdb`) that is connected to a server class using CNA cannot be opened by any other server class, including the default class.

The advantage of these restrictions is that because all operations on the database are known to be being performed on one node with pages resident in one cache, operations on the database do not require distributed locks, and the pages for the database do not need to participate in Distributed Multi-Cache Management (DMCM) protocols. For an installation servicing multiple databases, this allows you to increase efficiency by grouping your database operations by node, which significantly increases cache hit rates and decreases the latency of lock resolution and the overhead associated with DMCM. In addition, Row Level Locking and Update Mode Locks are automatically supported for databases serviced by CNA classes, instead of being silently converted to coarser granularity locks and stronger lock modes.

Appendix C: Configuring Ingres Cluster Solution for Linux

This section contains the following topics:

- [Ingres Cluster Solution Overview](#) (see page 86)
- [Requirements for the Ingres Cluster Solution](#) (see page 87)
- [Preparation for Installing the Ingres Cluster Solution](#) (see page 88)
- [How You Install the Ingres Cluster Solution for Linux](#) (see page 88)
- [How Client Applications Access an Ingres Cluster](#) (see page 90)
- [Class Node Affinity \(CNA\)](#) (see page 90)
- [Global File System and Database Creation](#) (see page 91)

Ingres Cluster Solution Overview

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

Ingres is installed in the typical manner, except most file locations must be on cluster file systems—that is, file systems using hardware and software that allow safe simultaneous update access from all machines intended to be part of the cluster. Certain other locations must *not* be on cluster file systems if the file system supports only block-oriented data transfer (for example, Oracle Cluster File System). Once installed, you run the `iimkcluster` utility to convert the initial instance into one of the cluster members (nodes), and then run the `iisunode` utility to add more nodes.

Use of the Ingres Cluster Solution requires specialized hardware and third-party software.

The Ingres Cluster Solution is incompatible with the following Ingres features:

- Row-level locking, except when accessing a database through a server class using “class node affinity”
- Update mode locks, except when accessing a database through a server class using “class node affinity”
- Two-phase commit (2PC)
- Auditing of tables with long data types

On each node, you should make sure that Ingres and your applications perform as expected. Some of the restrictions on lock level and lock mode are handled internally by Ingres, but may result in increased contention or deadlocks.

Requirements for the Ingres Cluster Solution

The Ingres Cluster Solution requires that you install Ingres enabled for Native POSIX Threads. To install Ingres enabled for Native POSIX Threads, you must be running a Linux distribution that provides Native POSIX Thread Library (NPTL) support. All Linux distributions based on the 2.6 kernel provide NPTL support, as does the Red Hat Enterprise Linux AS 3.0 2.4.21EL kernel.

In addition, you must have the following:

- A hardware cluster in which multiple machines can share the same physical storage. This cluster may take the form of a Storage Area Network (SAN), shared 1394b/"Firewire" drives, or shared Small Computer Systems Interface (SCSI) drives.
All disk locations must be set up as cluster file systems on storage accessible from the same path from all members of the cluster.
- A separate isolated private network for use by the Distributed Lock Manager and Quorum process.
- A supported cluster file system (CFS). This software allows simultaneous update access to the shared storage from multiple machines. For the latest information on supported CFS products, see <http://ingres.com>.
- The OpenDLM distributed lock manager, which can be obtained from the Ingres Corporation web site or from SourceForge at <http://sourceforge.net/projects/opendlm/>.
- The iiquorumd quorum management daemon, which can be obtained from the Ingres Corporation web site.

Preparation for Installing the Ingres Cluster Solution

Before installing Ingres in a Linux 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. Install any required non-Ingres cluster support software, and perform the procedures recommended by the provider for verifying correct installation. This software includes:
 - A cluster file system
 - OpenDLM
 - iiqorund
3. Plan your storage location layout as described for a stand-alone Ingres instance, with the restriction that your data (II_DATABASE), transaction log (II_LOG_FILE, II_DUAL_LOG), checkpoint (II_CHECKPOINT), journal (II_JOURNAL), and dump (II_DUMP) locations must all reside on a cluster file system.

How You Install the Ingres Cluster Solution for Linux

Follow these steps to install and configure Ingres in a Linux Cluster environment:

1. Install Ingres in a stand-alone configuration using the locations decided upon under Step 3 in Preparation for Installing the Ingres Cluster Solution (see page 88).
2. Verify that the stand-alone Ingres instance operates correctly.
Typically, it is easier to resolve any configuration issues at this stage because only one machine is in use. In addition, aside from the cluster file system support, Ingres is using operating system software only.
3. When you are confident that the stand-alone Ingres is operating correctly, shut down the instance. As the user that owns the instance, execute the `iimkcluster` utility, as follows:

```
iimkcluster
```

The utility prompts you for a node number and a nickname.

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 nickname is an optional simple alias, which you can use in any context in which you could specify a *nodename* parameter. The nickname appears in the error log in lieu of the machine name.

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, as well as directory *\$II_SYSTEM/ingres/admin/hostname* to which the *symbol.tbl* is relocated.

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. For each additional cluster member (node), set up your Ingres environment by sourcing the environment file. This file (*.ingXXbash* or *.ingXXtsch*) is located in *II_SYSTEM*.
7. Run the *iisunode* utility on each node. As the user that owns the instance, enter the following:

```
iisunode
```

The utility prompts you for a unique node number and nickname. 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 a private *symbol.tbl* file based on the initial nodes *symbol.tbl* file.
- Creates the transaction logs for this node.

8. 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

Class Node Affinity (CNA)

Ingres allows the creation of server classes that function as regular DBMS servers but can be configured for specialized situations. The server parameter `class_node_affinity`, if set for a server class, allows servers in this class to be started on only one node at a time.

The configuration name and server class name for the default CNA classes generated is `iicnann`, where `nn` is the node number, zero padded.

While `iimkcluster` and `iisunode` set up a separate CNA class for each node they are run on, these classes are not bound to the node they were defined on, but can be started on any node. In addition, any database (except `iidbdb`) that is connected to a server class using CNA cannot be opened by any other server class, including the default class.

The advantage of these restrictions is that because all operations on the database are known to be being performed on one node with pages resident in one cache, operations on the database do not require distributed locks, and the pages for the database do not need to participate in Distributed Multi-Cache Management (DMCM) protocols. For an installation servicing multiple databases, this allows you to increase efficiency by grouping your database operations by node, which significantly increases cache hit rates and decreases the latency of lock resolution and the overhead associated with DMCM. In addition, Row Level Locking and Update Mode Locks are automatically supported for databases serviced by CNA classes, instead of being silently converted to coarser granularity locks and stronger lock modes.

Global File System and Database Creation

The Red Hat Global File System (GFS) has poor performance using direct I/O and small page sizes, as required when using 2 KB pages with Ingres. (This problem was seen on GFS version 6.)

When creating databases with `II_DATABASE` on a GFS device, we recommend that you use the `createdb -page_size` option to specify a page size for your system catalogs of at least 4096 (4 KB). All tables residing in whole or in part on a GFS device should use page sizes of 4096 or higher. If you have work locations on GFS devices or want to avoid the need to explicitly specify a page size when creating a table, set the default page size (`default_page_size`) to 4096 or higher.

If the transaction logs reside in whole or in part on a GFS device, `block_size` should be at least 4.

Appendix D: System Requirements for OpenVMS

This section contains the following topics:

[System Hardware Requirements on OpenVMS](#) (see page 93)

[SYSGEN Parameters that Affect Operation](#) (see page 94)

[Recommended OpenVMS Process Resources](#) (see page 98)

[Required OpenVMS Privileges](#) (see page 100)

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 32 MB. However, the recommended size is between 100 and 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	For a DBMS Server installation:	
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> $\text{CHANNELCNT} = (\text{sessions} * 4) + \text{file_limit}$ <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>

Element	Description	Considerations
NPAGEDYN	Non-paged dynamic pool; determines the number of bytes to allocate for the non-pages dynamic pool	Used in allocating the logging and locking system in previous Ingres releases. Ingres uses GBLPAGES (global pages) rather than NPAGEDYN and NPAGEVIR to allocate the logging and locking system shared memory at startup time.
NPAGEVIR	Virtual non-paged pool; determines the number of bytes to which the non-paged pool may be extended	Used in allocating the logging and locking system in previous Ingres releases. Ingres uses GBLPAGES (global pages) rather than NPAGEDYN and NPAGEVIR to allocate the logging and locking system shared memory at startup time.
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>

Element	Description	Considerations
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>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>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>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
TQUELM	Timer queue entry limit	40
PGFLQUO	Paging file quota (possibly set to the system virtual page count)	250,000
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)	2048
JTQUOTA	Job logical name table byte limit	20,000

UAF Name	Description	Recommended Starting Value
MAXDETACH	Maximum detached processes limit	0 (infinity)
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

OpenVMS quotas and their corresponding Ingres parameters are listed here. Adjust the Ingres parameters using Configuration-By-Forms to reflect the OpenVMS quota settings.

Ingres Configuration	OpenVMS (Authorize) Entry
vms_ast_limit	ASTIm
vms_buffer_limit	BYTIm
vms_extent	Wsextent
vms_file_limit	Fillm
vms_io_buffered	BIOIm
vms_io_direct	DIOIm
vms_maximum_working_set	Wsquo
vms_page_file	Pgflquo
vms_privileges	DEFPRIVILEGES
vms_queue_limit	Enqlm
vms_working_set	Wsdef

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).

Index

-
- .NET Data Provider • 29
- A**
 - access
 - the instance • 34
 - to database and tools • 62, 77, 78
 - using shadow passwords • 62
 - accessdb command • 62, 77
 - automatic startup • 61, 77
- B**
 - Bridge Server • 19
- C**
 - catalogs • 75
 - character set • 17
 - checkpoint
 - files • 14
 - class node affinity • 83
 - client installation • 13, 47
 - client-only disk configuration • 13
 - cluster
 - VMS • 69, 73
 - Communications Server • 19, 46, 47
 - computer names • 20
 - Configuration-By-Forms • 41
 - configurations, major • 46
 - customization after installation • 41
- D**
 - Data Access Server • 19
 - database
 - files • 14
 - DBMS Server
 - installation parameters • 14
 - networked • 46, 47
 - disk configurations • 12, 13
 - documentation • 29
 - domain • 8
 - dump files • 14
- E**
 - environment variables
- II_CHECKPOINT • 14
- II_DATABASE • 14
- II_DUMP • 14
- II_JOURNAL • 14
- II_LOG_FILE_SIZE_MB • 14
- II_SYSTEM • 49, 56, 75
- II_TIMEZONE_NAME • 15
- II_WORK • 14
- F**
 - file locations • 59
- H**
 - history log file • 71
- I**
 - II_CHECKPOINT • 14
 - II_DATABASE • 14
 - II_DISTRIBUTION • 58
 - II_DUMP • 14
 - II_JOURNAL • 14
 - II_SYSTEM • 49, 56, 72, 75
 - II_TIMEZONE_NAME • 15
 - II_WORK • 14
 - ingbuild command • 50
 - ingmknfs command • 54
 - ingsetenv command • 41
 - ingstart and ingstop commands • 39
 - installation
 - history log file • 79
 - owner • 8, 49
 - parameters, general • 13
 - installing Ingres
 - on UNIX • 49, 50
 - on Windows • 29
- J**
 - journal files • 14
- L**
 - Linux
 - installing Ingres on • 21
 - location • 59
 - log file
 - as a raw device • 58

logging system
 making changes to • 41
logicals
 concealed • 72, 73

M

multiple instances • 8, 71

N

Name Server • 19, 47
names • 20
netutil utility • 41
network protocols • 19
NFS client • 13, 47, 54, 56

O

OpenVMS
 hardware requirements • 93
 process resources • 98
 quotas • 99
 required privileges • 100
 SYSGEN parameters • 94
 system requirements • 93

P

parameters
 DBMS Server installation • 14
 general installation • 13
password
 DBMS Server • 41
 shadow • 62

Q

quick start • 41

R

raw log file • 58, 59
Recovery Server • 19
response file • 23, 24, 25, 26, 30, 50
RPM • 21, 22, 23

S

servers
 Ingres • 19
Service Manager (Windows) • 38
setenv command • 57, 58
setup program • 75
shadow passwords • 62

Star Server • 19
start Ingres • 35, 36, 37, 38, 39
startup, automatic • 61, 77
SYSGEN parameters • 74, 94
system administrator account • 8
 create • 46

T

TERM_INGRES • 57, 69
terminal
 define • 57, 69
time zone • 15
transaction log file
 as a raw device • 58

U

UIC • 74
uninstall Ingres • 43, 44
upgrading • 49, 75
user
 access to tools • 62, 78
 login file • 62, 63
 names • 20

V

Vision • 29
Visual Manager • 37

W

work files • 14