Informix Product Family
Informix
Version 12.10

IBM Informix Installation Guide

SC27-8018-00

Informix Product Family
Informix
Version 12.10

IBM Informix Installation Guide

SC27-8018-00

Note
Before using this information and the product it supports, read the information in “Notices” on page B-1.

Edition

This edition replaces GC27-4527-03 and GC27-4526-02.

Contents

Introduction

About this publication

This publication describes how to install, configure, and initialize an IBM® Informix® database server.

Types of users

Typically, database administrators (DBAs) install an Informix server. The documentation assumes that you are familiar with the operating procedures of your computer and with your operating system.

What's new in installation for IBM Informix, Version 12.10

This publication includes information about new features and changes in existing functionality.

For a complete list of what's new in this release, go to http://www.ibm.com/support/knowledgecenter/SSGU8G_12.1.0/com.ibm.po.doc/new_features_ce.htm.

Table 1. What's new in installation for IBM Informix, Version 12.10.xC5

Overview	Reference
Support for Java™ 7 IBM Informix 12.10.xC5 software supports Java Platform Standard Edition (Java SE), Version 7. Informix installation applications install IBM Runtime Environment, Java Technology Edition, Version 7 on most platforms by default. That version is used to run Java user-defined routines that are created in the server. Check the machine notes for your operating system platform to determine whether the installation application requires a preinstalled JRE.	"System requirements" on page 1-3
Improved installation logging and debugging The default name and location of the installation log file for both the database server and client products is now `/tmp/iad_act.log` (UNIX, Linux, Mac OS X) or `\tmp\iad_act.log` (Windows). You can specify a different name and location for the installation log file with the **-DLOG_FILE** option in the installation command. The **-DDEBUG** option is deprecated. Now you have more control over the debugging and tracing information for the installation process. By default, tracing is disabled. You can set the tracing level 1 - 9 with the **-DDEBUG_LEVEL** option in the installation command. If tracing is enabled, the default name and location of the debugging file is `/tmp/iad_dbg.log` (UNIX, Linux, Mac OS X) or `\tmp\iad_dbg.log` (Windows). You can set the name and location of the debug file with the **-DDEBUG_FILE** option.	Chapter 5, "Installation troubleshooting," on page 5-1 Chapter 4, "Installation command syntax," on page 4-1

Table 1. What's new in installation for IBM Informix, Version 12.10.xC5 (continued)

Overview	Reference
Easier silent installations You can streamline a silent installation on systems with existing installations of the database server or client products. Include the **-DOVERWRITE_PRODUCT=TRUE** option with the **ids_install**, **installclientsdk**, or **installconnect** command to overwrite an existing installation. Otherwise, the installation application exits if it finds an existing product installation.	Chapter 4, "Installation command syntax," on page 4-1
Deprecated options for installation commands The following options to the **ids_install**, **installclientsdk**, and **installconnect** commands are deprecated: • UNIX, Linux: The **-i swing** installation command option is deprecated. Use the **-i gui** option in the installation command to run the installation application in GUI mode. • The **-DDEBUG** option is deprecated. Use the new **-DDEBUG_LEVEL** option in the installation command to set the tracing level.	Chapter 4, "Installation command syntax," on page 4-1

Table 2. What's new in installation for IBM Informix, Version 12.10.xC4

Overview	Reference
Uninstall Informix programs on Windows from the Control Panel Windows users with administrator privileges can uninstall Informix products by using the Control Panel. Previous methods of modification and uninstallation are still available.	"Removing or modifying installations (Windows)" on page 10-2
Easier installation of 32-bit programs on Windows 64-bit operating systems When you install Informix products from 32-bit installation media on 64-bit Windows computers, you no longer need to add the `SysWOW` compatibility folder to the **`PATH`** environment variable.	

Table 3. What's new in installation for IBM Informix, Version 12.10.xC3

Overview	Reference
Automatically configure the server during installation If you create a server during installation, the server is configured based on your selections in the installation program, storage spaces are created, automatic tuning of resources for performance is enabled, and the JSON wire listener is started.	"Database server configuration during installation" on page 1-10

Java technology dependencies

IBM Informix software supports Java Platform Standard Edition (Java SE) to create and run Java applications, including user-defined routines (UDRs). Java SE 7 is supported as of Informix 12.10.xC5, while Java SE 6 is supported in earlier fix packs.

Important:

- Check the machine notes to learn about Java technology exceptions and other requirements for specific operating system platforms. The machine notes are available on the product media and in the online release information.
- In general, any application that ran correctly with earlier versions of Java technology will run correctly with this version. If you encounter problems, recompile the application with the next available fix pack or version. However, because there are frequent Java fixes and updates, not all of them are tested.
- To develop Java UDRs for the database server, use the supported Java software development kit or an earlier version according to Java compatibility guidelines. The supported version provides a known and reliable Java environment for UDRs in this database server release.

For details about Java requirements, check the following sections:

"Java runtime environment"

"Software development kit for Java"

"Java Database Connectivity (JDBC) specification" on page viii

Java runtime environment

On most supported operating system platforms, the Informix installation application bundles a Java runtime environment that it requires. However, check the machine notes for your operating system platform to determine whether the installation application requires a particular Java runtime environment to be preinstalled.

Also, IBM Runtime Environment, Java Technology Edition is supported for general use of the database server. It is installed on most operating system platforms by default in the following directory: `$INFORMIXDIR/extend/krakatoa/jre/`.

MongoDB API and REST API access supports IBM Runtime Environment, Java Technology Edition, Version 7.

Software development kit for Java

The following products and components require a software development kit for Java, but one is not installed:

- Informix DataBlade® Developers Kit (DBDK)
- IBM Informix JDBC Driver
- J/Foundation component
- Spatial Java API
- TimeSeries Java API

The software development kit that you use must be compatible with the supported Java runtime environment. Informix does not support OpenJDK. You can download a development kit from the following web sites:

- **Recommended for AIX and Linux:** IBM SDK, Java Technology Edition (http://www.ibm.com/developerworks/java/jdk/)
- **Recommended for HP-UX:** HP-UX 11i Java Development Kit for the Java 2 Platform Standard Edition (https://h20392.www2.hp.com/portal/swdepot/displayProductInfo.do?productNumber=HPUXJAVAHOME)
- Oracle Java Platform, Standard Edition Development Kit (JDK) (http://www.oracle.com/technetwork/java/javase/downloads/index.html)

Java Database Connectivity (JDBC) specification

Informix products and components support the Java Database Connectivity (JDBC) 3.0 specification.

Example code conventions

Examples of SQL code occur throughout this publication. Except as noted, the code is not specific to any single IBM Informix application development tool.

If only SQL statements are listed in the example, they are not delimited by semicolons. For instance, you might see the code in the following example:

```
CONNECT TO stores_demo
...

DELETE FROM customer
   WHERE customer_num = 121
...

COMMIT WORK
DISCONNECT CURRENT
```

To use this SQL code for a specific product, you must apply the syntax rules for that product. For example, if you are using an SQL API, you must use EXEC SQL at the start of each statement and a semicolon (or other appropriate delimiter) at the end of the statement. If you are using DB–Access, you must delimit multiple statements with semicolons.

Tip: Ellipsis points in a code example indicate that more code would be added in a full application, but it is not necessary to show it to describe the concept that is being discussed.

For detailed directions on using SQL statements for a particular application development tool or SQL API, see the documentation for your product.

Additional documentation

Documentation about this release of IBM Informix products is available in various formats.

You can access Informix technical information such as information centers, technotes, white papers, and IBM Redbooks® publications online at http://www.ibm.com/software/data/sw-library/.

Compliance with industry standards

IBM Informix products are compliant with various standards.

IBM Informix SQL-based products are fully compliant with SQL-92 Entry Level (published as ANSI X3.135-1992), which is identical to ISO 9075:1992. In addition, many features of IBM Informix database servers comply with the SQL-92 Intermediate and Full Level and X/Open SQL Common Applications Environment (CAE) standards.

How to read the syntax diagrams

Syntax diagrams use special components to describe the syntax for SQL statements and commands.

Read the syntax diagrams from left to right and top to bottom, following the path of the line.

The double right arrowhead and line symbol ▸▸— indicates the beginning of a syntax diagram.

The line and single right arrowhead symbol —▸ indicates that the syntax is continued on the next line.

The right arrowhead and line symbol ▸— indicates that the syntax is continued from the previous line.

The line, right arrowhead, and left arrowhead symbol —▸◂ symbol indicates the end of a syntax diagram.

Syntax fragments start with the pipe and line symbol |— and end with the —| line and pipe symbol.

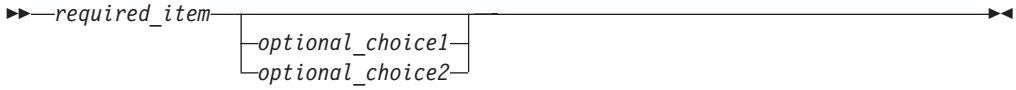
Required items appear on the horizontal line (the main path).

Optional items appear below the main path.

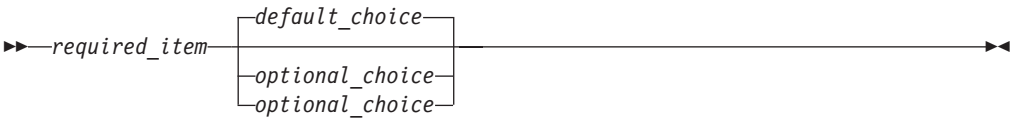
If you can choose from two or more items, they appear in a stack.

If you *must* choose one of the items, one item of the stack appears on the main path.

If choosing one of the items is optional, the entire stack appears below the main path.

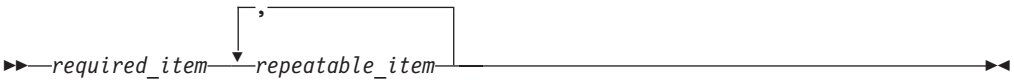


If one of the items is the default, it will appear above the main path, and the remaining choices will be shown below.



An arrow returning to the left, above the main line, indicates an item that can be repeated. In this case, repeated items must be separated by one or more blanks.

If the repeat arrow contains a comma, you must separate repeated items with a comma.

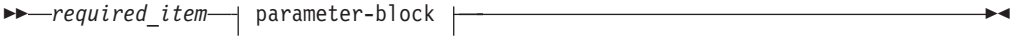


A repeat arrow above a stack indicates that you can make more than one choice from the stacked items or repeat a single choice.

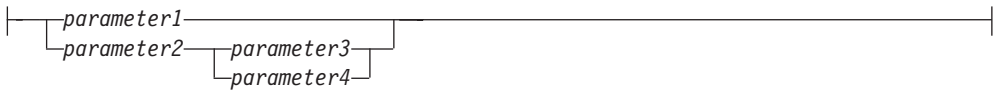
SQL keywords appear in uppercase (for example, `FROM`). They must be spelled exactly as shown. Variables appear in lowercase (for example, `column-name`). They represent user-supplied names or values in the syntax.

If punctuation marks, parentheses, arithmetic operators, or other such symbols are shown, you must enter them as part of the syntax.

Sometimes a single variable represents a syntax segment. For example, in the following diagram, the variable `parameter-block` represents the syntax segment that is labeled **parameter-block**:



parameter-block:



How to provide documentation feedback

You are encouraged to send your comments about IBM Informix product documentation.

Use one of the following methods:

- Send email to docinf@us.ibm.com.

- Add comments to topics directly in IBM Knowledge Center and read comments that were added by other users. Share information about the product documentation, participate in discussions with other users, rate topics, and more!

Feedback from all methods is monitored by the team that maintains the user documentation. The feedback methods are reserved for reporting errors and omissions in the documentation. For immediate help with a technical problem, contact IBM Technical Support at http://www.ibm.com/planetwide/.

We appreciate your suggestions.

Chapter 1. Preparing for installation

Before you install Informix database server products, you must prepare your system and plan for choices that you must make during the installation process.

Preparing for installation includes reviewing system requirements, planning how to upgrade, deciding on the installation owner and the installation directory, reviewing installation options, and gathering information if you want to create a database server during installation. It also includes preinstallation tasks such as obtaining the software and reviewing the machine notes.

To prepare for installation:

1. Obtain the software. You can download the product package from Passport Advantage: http://www.ibm.com/software/howtobuy/passportadvantage.
2. Review the Informix components that can be installed with the database server. You can install all of them or some of them.
3. Ensure that your computer meets the system requirements.
4. Choose a user account with which to install the product. The user account becomes the installation owner.
   - UNIX, Linux, Mac OS X: Decide between a non-root owner and an owner with root or administrative privileges.
   - If you plan to install as an administrative user, decide on a password for user **informix**.
5. Choose an installation directory. If the database server is already installed, decide on the upgrade path.
6. Decide between a typical or custom installation.
   a. Choose a typical installation in the following cases:
      - To install all database server components, IBM Informix Client Software Development Kit, and the IBM Informix JDBC Driver.
      - To create a configured database during installation with connectivity for JSON applications and multiple storage spaces for data.
   b. Choose a custom installation if you want to have any of the following options:
      - To install a subset of database server components.
      - To choose which client products to install.
      - In root-based installation, to set up event auditing for the security policies of your organization.
      - To create a configured database server during installation with one storage space for data.
      - To prepare a response file for a non-interactive installation.

      Depending on your operating system, you can choose other types of installations that have similar options to a custom installation. UNIX, Linux, Mac OS X: You can choose the Legacy installation to prepare a script-based installation. Linux: You can choose the RPM installation to generate an RPM package.
7. Decide whether to create a configured database server during installation.

- If you do not create a configured database server during installation, you must configure the database server and initialize disk space after installation is complete.
- If you create a database server during installation, you must supply some information:
  - Specify information to configure the database server, such as the number of expected users.
  - Windows: For a typical setup, specify a password for user **ifxjson**. This user administers the JSON wire listener.

Log files are created during the installation process. You can use the log files to troubleshoot installation errors.

After you install the database server, you can create the demonstration databases that are included with the database server. Many examples in the documentation and in the `$INFORMIXDIR/demo` (UNIX, Linux, Mac OS X) or the `%INFORMIXDIR%\demo` (Windows) directory are based on the demonstration databases. You can use the demonstration databases to explore database server features. You can create the demonstration databases at any time after installation by running the **dbaccessdemo** command. See Demonstration databases.

Informix components

When you install the IBM Informix database server, you can also install related products, such as client APIs and other IBM products.

The following diagram provides an overview of the base architecture of the Informix database server and related products.

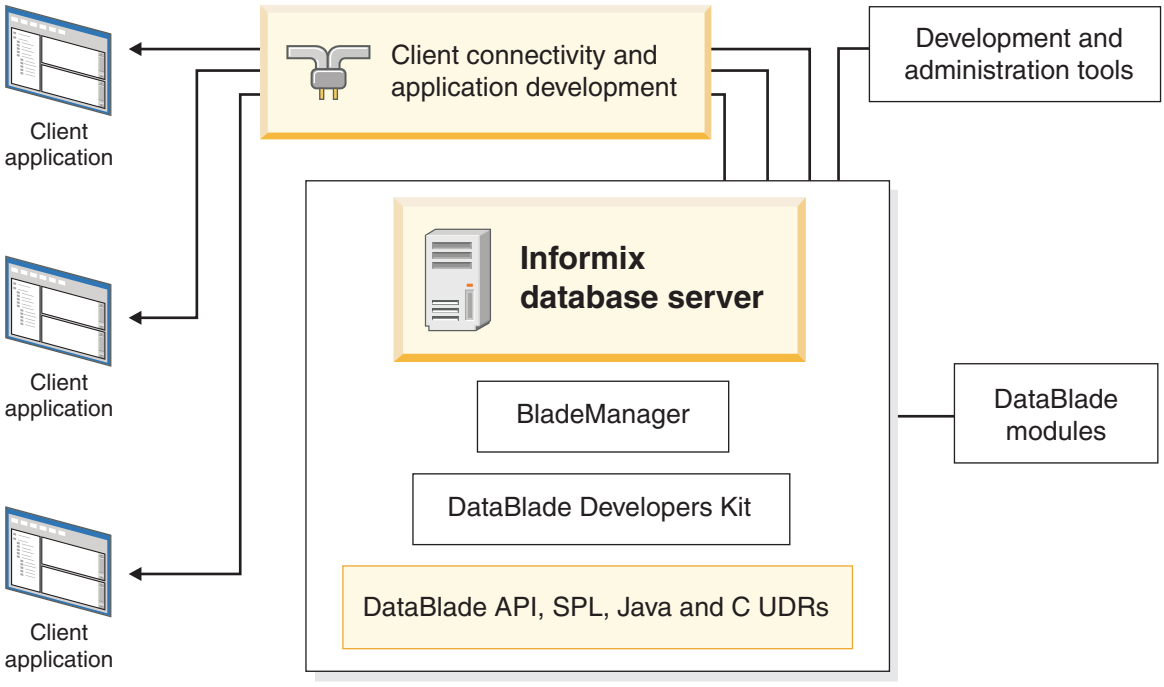


Figure 1-1. Architecture of Informix database server and related products

Depending on your edition of Informix and your operating system, the product package includes installation programs for some or all of the following associated products:

IBM Informix Client Software Development Kit and IBM Informix Connect
Contains client APIs for developing and running client applications, and contains the IBM OpenAdmin Tool (OAT) for Informix for monitoring and managing the database server. IBM Informix Connect contains only the runtime libraries of the client APIs to allow applications that run on client computers to access the database server.

Informix DataBlade Developers Kit (DBDK) (Windows)
Contains tools for developing and packaging DataBlade modules.

IBM Informix JDBC Driver
Java interfaces and classes to connect to Informix databases.

IBM Informix BladeManager
A utility to register and unregister DataBlade modules.

IBM Informix Web DataBlade Module
Tags and functions to create web applications that incorporate data that is retrieved dynamically from the Informix database.

International Language Supplement
The core Global Language Support locale files.

IBM Data Studio
An integrated development environment to develop and test SQL and XQuery queries, stored procedures, web services, and Java data access layers.

IBM Cognos® Business Intelligence
A web-based business intelligence solution with integrated reporting, analysis, scorecarding, and event management features.

IBM SPSS® Statistics Desktop and Data Drivers
A set of tools for statistical analysis and drivers for various data sources.

IBM SPSS Modeler
A set of data mining tools to develop predictive models and deploy them into business operations to improve decision making.

Related information:

Product versions

System requirements

Before you install the Informix database server, make sure that your computer meets the system requirements.

Operating system requirements

The Informix database server system requirements web page lists supported operating systems: http://www.ibm.com/support/docview.wss?uid=swg27013343.

Install the required operating system patches and library files that are listed in the machine notes for your operating system. The machine notes are in therelease information or in the `doc` directory of the media files before installation.

UNIX, Linux, Mac OS X: The directory on which you plan to install the database server must be local or an NFS-mounted file system with regular operating-system files.

Windows: The drive on which you plan to install the database server must be formatted with NTFS and must have 8.3 file name creation enabled. For Windows 7 and subsequent Windows versions, turn off User Account Control (UAC) security before you install the database server.

Disk space and memory requirements for installation

The following disk space and memory requirements for installing the database server are approximate. The requirements for some operating systems might be lower.

Disk space
Approximately 750 MB of disk space is required for a typical installation.

Some installation choices require more disk space. The installation application informs you of the total disk space that is required by your setup before you copy the binary files to your host computer.

RAM Approximately 1 GB RAM is required.

Temporary disk space
You can change the location of the temporary directory by setting the **IATEMPDIR** environment variable before you run the installation application.

UNIX, Linux, Mac OS X: 2 GB free space is required in your temporary directory (`/tmp` by default). If the temporary directory does not have enough space, the installation application attempts to write the information into the home directory of the user account.

Windows: 1 GB free space is required in your temporary directory (`C:\tmp` by default).

Java runtime environment (JRE) requirements

The installation application bundles a Java runtime environment that it requires. However, check the machine notes for your operating system platform to determine whether the installation application requires a preinstalled JRE.

Related reference:

Java technology dependencies

Installation owner

The owner of the Informix database server depends on the privileges of the user who runs the installation application. The owner of the server creates and controls all other user accounts that can access the database server.

The standard way to install the database server is as a superuser with administrative privileges. The installation application creates the user **informix**. User **informix** is a user account with main authority over the database server instance.

UNIX, Linux, Mac OS X: If you install the database server as a non-administrative, or non-root, user, you become the owner of the database server. In this case, user

informix in not created. You cannot change the owner of the database server. You cannot convert or upgrade a non-root installation to a standard, root-based installation, and vice versa.

User informix

User **informix** is required for root-based installations because it has the unique user identifier (UID) to manage and maintain database server instances and databases on the host server.

UNIX, Linux, Mac OS X: User **informix** is a member of the group **informix**. On UNIX or Linux, if group **informix** exists on your system, but user **informix** does not exist, you must create user **informix** before you install the database server.

Windows: User **informix** is a member of the Informix-Admin group and the Administrators group. In most cases, the installation application automatically creates the user **informix** and the group **informix** or Informix-Admin group. User **informix** can log on as a service and act as part of the operating system.

If you are installing the database server for the first time on your system, the installation application prompts you to provide a password for the **informix** user. If user **informix** exists on your system, the installation application prompts you to confirm the password.

The password for the **informix** user account must be protected. Let only trusted database and security administrators log in as user **informix**.

Important: The database server does not start if password standards for user **informix** or any other users do not conform to local security standards.

UNIX, Linux, Mac OS X: Non-administrative, or non-root owner

If you run the installation application as a non-administrative user, that user is the *non-root owner* of the installation and has database server administrator (DBSA) privileges over the database server.

The installation directory must be on a local file system if you plan to use `onipcstr` connections because you cannot create a named domain socket on a non-local file system.

A non-root installation is appropriate in the following situations:

- You plan to embed the database server in an application that is to be deployed on other computers where you either do not want a root-level installation or you want non-administrative users to install the application.
- You do not have or do not want to use root user credentials for the Informix database server installation.
- You are developing a virtual appliance.

If you install the database server without root privileges, you cannot use the following features and tools:

- Enterprise Replication for servers that have different owners
- High-availability clusters
- Auditing through role separation
- The **ON-Bar** utility for backing up and restoring data

- The **onperf** utility to monitor database server performance

Encrypted connections and column-level encryption (CLE) are supported only if the required security-layer plug-in is installed separately on the host computer. You can run the IBM Global Security Kit (GSKit) installation application in the `INFORMIXDIR/gskit` directory to set up the security layer.

Creating the group informix and user informix (UNIX, Linux)

In a few situations during a root-based installation, you must create the group **informix** and user **informix** before you install the database server on UNIX or Linux operating systems.

You must have root authority to create users or groups.

You must create the objects before you start the installation application in the following situations:

- You want to specify a particular identifier (ID) number.
- The group **informix** exists on the system; however, the user **informix** does not. In this case, you must create only user **informix**.

To create the objects:

1. If necessary, create the group **informix** by using the following command, where *n* is a unique identifier (ID) greater than 100:
   - AIX: `mkgroup -a n informix`
   - UNIX, Linux, Mac OS X: `groupadd -g n informix`
2. Create the user **informix** by using the **useradd** command, where *n* is a unique identifier (ID) greater than 100:
   - `useradd -u n -g informix informix`

   Important: Add users to the group **informix** only if the users need administrative access to the database server.
3. Create a password for user **informix** by using the **passwd** command.

Database server upgrades

If you have an earlier version of the Informix database server installed, you must plan and prepare your system before you upgrade to a new major version or a fix pack of the same version.

The *IBM Informix Migration Guide* explains how to plan for and perform the required prerequisite tasks for upgrading to a new version of the product, and how to apply fix packs and interim fixes. Review those topics carefully before you install the new software.

Related concepts:

"Installation directory" on page 1-7

Related tasks:

Preparing for migration (Migration Guide)

Installation directory

You can use the default installation directory for the database server when you run the installation application, or you can create a directory before you install and then select that directory during installation.

The installation directory is known as the INFORMIXDIR directory because the environment variable **INFORMIXDIR** is set to the installation directory.

The directory where you install Informix products must fulfill these requirements:

- The directory must be empty, except in the following situations:
  - You are upgrading to a fix pack of the same version.
  - You accept the risks of upgrading to a new version by replacing the existing version.
- The full path to the installation directory must not exceed 200 characters, including path separators, and must not contain spaces. UNIX, Linux, Mac OS X: The directory for a non-root installation must not exceed 60 characters.
- The installation path, including all path elements, must be secure before the database server can start.

Important: The database server will not start unless the installation directory is secure.

UNIX, Linux, Mac OS X: If the installation application finds a security issue with the installation directory, you must choose one of the following methods to secure the directory:

Let the installation program secure the path (recommended)
The installation continues and generated script fixes nonsecure directories along the installation path.

Continue installation; manually run script to secure the path later
After installation is complete, you must run the **onsecurity** utility to generate a security script, and then run the security script. The **onsecurity** utility is in the /SERVER/ directory on the installation media. After installation, you can find the utility in the $INFORMIXDIR/bin directory.

View other options to make the installation directory secure
You must select how to reset the permissions for the owner, group, and installation directory. You must understand what constitutes a secure installation path in your environment, including the requirements of user **informix** and group **informix**.

Related concepts:

IBM Informix directory security (Security Guide)

"Database server upgrades" on page 1-6

Related tasks:

Securing $INFORMIXDIR and its subdirectories (Security Guide)

Installation options

The primary way to install the Informix database server is with an interactive installation that prompts you to configure installation properties. After your run an interactive installation, you can run non-interactive installations that are based on the original installation.

Interactive database server installation setup options

An interactive database server installation has the following setup options:

Typical installation
Recommended. You install the database server with all associated components, IBM Informix Client Software Development Kit, and the IBM Informix JDBC Driver.

Custom installation
You can minimize the size of the database server on disk by excluding unneeded features and products.

You can choose which client products to install. To determine whether you need to install client products, see *IBM Informix Client Products Installation Guide*.

You can set up auditing through role separation.

You can prepare for a silent installation by creating a response file.

UNIX, Linux, Mac OS X: Legacy installation
You have the same options as a custom installation, plus the product files are extracted to create a script-based non-interactive installation.

Linux: RPM installation
You have the same options as a custom installation, plus an RPM image is generated to create an RPM non-interactive installation.

For all setup options, you have the following choices:

- Whether to create a configured database server during installation. The database server that is created during installation differs slightly for a typical or custom installation. If you do not create a database server during installation, you must manually configure and start the database server after installation.
- The location of the installation directory.
- The identity of the installation owner.

Non-interactive installation methods

Non-interactive installation methods, such as a silent installation or a script-based installation, are based on the results of interactive installation or on an existing database server installation.

Event auditing

If you choose a custom setup, you can enable event auditing. Event auditing tracks selected activities that users perform. You can improve the security of your event-auditing procedures by configuring role separation, which provides members of certain group identifiers (group IDs) on your system the privileges to manage and examine auditing records. Role separation provides increased database security because the database server separates administrative tasks into mutually exclusive roles.

Important:

- Role separation is not supported in a non-root installation.
- You must select custom installation setup to enable role separation.

- If you enable role separation, you cannot turn it off after the database server is installed. To remove role separation, you must uninstall the database server and reinstall it without role separation.

UNIX, Linux, Mac OS X: Role separation

If you do not enable role separation, the **informix** group has privileges to perform all administrative tasks.

Role separation provides two roles:

Database System Security Officer (DBSSO)
Controls what the auditing subsystem monitors and which actions database users can perform.

Auditing Analysis Officer (AAO)
Controls whether auditing occurs, maintains the audit log files, and analyzes the audit records.

The **informix** group is the default group that is associated with the two roles. During installation, you can replace the default groups with existing groups.

After installation is complete, establish an audit-only user account for each individual who acts as a DBSSO or AAO. For example, a person with DBSSO responsibilities can have the user **DBSSO1** account, and also have the user **garcia5** account for general database server access.

Windows: Role separation

If you do not enable role separation, the **Informix-Admin** group has privileges to perform all administrative tasks.

If you enable role separation during installation, you are prompted to create groups and users and add the users to the corresponding groups. During installation, you can replace the default users and groups with existing users or groups.

Table 1-1. Role separation

Header	Header	Header
Informix-Admin	General Database Administration	Performs general administrative tasks, such as archiving and restoring data, monitoring use and performance, and tuning the system.
ix_dbsso	Database System Security Officer	Maintains the security of the database server. Functions of this role include audit adjustment and changing security characteristics of storage objects. Creation of this user role requires selection of a password during installation.

Table 1-1. Role separation (continued)

Header	Header	Header
ix_aao	Auditing Analysis Officer	Audits the records of specific types of database activities. If someone attempts to circumvent or corrupt the security mechanism of the database, these actions can be traced. Creation of this user role requires selection of a password during installation.
ix_users	Database Users	Accesses the database to perform user tasks. Only users who are designated as members of the **ix_users** group can access the database.

Related concepts:

Secure-auditing facility (Security Guide)

Exclude database server components

If you choose a custom setup, you can minimize the size of the database server on disk by excluding some components. Reducing the size of the installation can be useful if you plan to embed the database server in other systems or applications.

The Informix database server consists of discrete, separately installable components. You can select to install only the database server components that your application and deployment require. For example, you might not need replication capabilities. Some components are mutually dependent, and must coexist in the instance. The installation application enforces these dependencies by automatically including dependent components or informing you when a combination of selections is not supported.

You can easily modify your installation by adding or removing separately installable components after the database server is installed without installing the base database server again. Adding or removing components after you installed the database server does not affect the integrity of your system. However, all Informix components must run on the same version as the core database server.

Database server configuration during installation

You can choose to create and configure a database server during installation. The database server is configured by the choices that you make in the installation application. After installation is complete, disk space for the database server is initialized and the database server is started.

All mandatory configuration parameters and environment variables are set and default connectivity information is configured. The database server is also configured to require minimal administration. The database server is configured to automatically tune resources for performance, which include expanding the buffer

pool, the logical logs, and the physical log, and increasing virtual processors. You can further customize the database server by setting other configuration parameters and environment variables.

Important: If you plan to use a locale or language other than the default (US English), set the appropriate global language support (GLS) environment variables before you create a database. For example, to prevent server and collation problems, it is crucial to set the GL_USEGLU environment variable correctly. The same Unicode collation must be used for the database and the client application environment.

The following storage spaces are created for an installation of the database server:

- A root dbspace to contain control and tracking information
- An extendable plogspace for the physical log
- A dbspace for the logical logs
- A temporary dbspace
- An sbspace
- A temporary sbspace
- Dbspaces for data:
  - Typical installation: Three extendable dbspaces with the default page size, and three extendable dbspaces with an 8 KB page size
  - Custom installation: One dbspace with the default page size

The sizes of the spaces are based on the number of expected users that you specify during the installation.

If you install IBM Informix Client Software Development Kit (Client SDK) as part of the database server installation, IBM OpenAdmin Tool (OAT) for Informix is configured and ready to connect to the server.

If you run a typical installation, the server is configured with a JSON wire listener for connections to MongoDB applications. The JSON wire listener is started, and connected to the database server through the **ifxjson** user.

Windows: A program group for the database server is created from which you can open a command prompt for the server, start the Server Instance Manager, uninstall the database server, or view documentation.

Related concepts:

Install the wire listener (JSON compatibility)

Overview of database server configuration and administration (Administrator's Guide)

Related reference:

GLS-related environment variables (GLS User's Guide)

Chapter 2. Running an interactive installation

You can install the Informix database server by running the interactive installation application. You can choose a typical or a custom setup.

Important: If you want to create a response file to use as a template for installing the database server on other computers, follow the procedure in "Preparing a response file" on page 3-1.

Before you install the database server, complete the following tasks:

- Prepare your system for installation.
- Log in as the appropriate user to be the owner of the database server.

You can include options to the installation command if you run the command from the command line. For example, you can change the name or location of the installation log file.

To install Informix:

1. Start the installation application from the media directory.
   - UNIX, Linux:
     a. If necessary, extract the product files. For example, run the **tar** command:
        `tar xvf filename`

        The *filename* is the name of the product `tar` file.
     b. Run the following command: **ids_install**
   - Mac OS X:
     a. If you do not see the `ids_install.app` icon when you open the installation media, double-click the self-extracting `.dmg` icon.
     b. Double-click the `ids_install.app` icon.
     c. If you want a non-root installation, select the **Private Installation** option.
   - Windows: Double-click the `ids_install.exe` file.
2. Select either **Typical** or **Custom** setup and follow the instructions in the installation application.
3. Complete the installation and exit the installation application.

Log files are created during the installation process. You can use the log files to troubleshoot installation errors.

If you created a database server during installation, the database server is configured, initialized, started, and ready for client connections. To see the connectivity information, look at the `sqlhosts` file. To see the configuration information, look at the `onconfig` file. Both files are in the `$INFORMIXDIR/etc` (UNIX, Linux, Mac OS X) or the `%INFORMIXDIR%\etc` (Windows) directory. You can customize the database server properties by setting configuration parameters in the `onconfig` file, setting environment variables in your environment, and adding connectivity information to the `sqlhosts` file. See Overview of database server configuration and administration.

If you did not create a database server during installation, you must configure and initialize the database server.

Related concepts:

Chapter 5, “Installation troubleshooting,” on page 5-1

Related reference:

Chapter 4, “Installation command syntax,” on page 4-1

Chapter 3. Running non-interactive installations

You can run non-interactive installations after you first run an interactive installation during which you create an installation template.

A silent, or unattended, installation requires that you create a response file, copy the product installation files and the response file to the target computer, and then run the silent installation command. The response file acts as a template for the silent installation.

UNIX, Linux, Mac OS X: A scripted installation requires that you copy an existing product installation directory to the target computer and run a script.

If you use Enterprise Replication or high-availability clusters, you can create a server clone from a snapshot of an existing database server with the **ifxclone** utility.

If you plan to embed Informix products in your application, you can use one of the following methods to deploy the database server:

- You can use the deployment utility to deploy snapshots of preconfigured database servers on one or more computers. A snapshot is an image of a database server that includes the installation directory, configuration settings, and any data spaces that are associated with the instance. The installation can be a working instance, or an installation that you set up as a template from which to deploy the instance on other computers. You can use the deployment assistant to customize the snapshot.
- Linux: You can run an RPM installation to create an RPM image of a customized database server installation, and then deploy the image through a silent installation.

Related concepts:

The deployment utility (Embeddability Guide)

Related tasks:

Deploying Informix and client products with RPM-based installation (Linux) (Embeddability Guide)

Related reference:

The ifxclone utility (Administrator's Reference)

Preparing a response file

Before you run a silent installation, you must run an interactive installation to create a response file. The response file contains information about how to install the database server on other computers that have the same basic type of operating system: either a UNIX operating system (including Linux and Mac OS X), or a Windows operating system.

Before you start the interactive installation, complete the following tasks:

- Prepare your system for installation.
- Log in as the appropriate user to be the owner of the database server.

The response file contains installation settings for a product and its features. You have the following choices for the response file:

Default response file
The default response file, `bundle.properties`, produces a typical installation. The `bundle.properties` file is in the top directory of the installation media. You must either edit the `bundle.properties` file to set the `LICENSE_ACCEPTED` property to `TRUE`, or include the **-DLICENSE_ACCEPTED=TRUE** option in the silent installation command.

Response file that is generated by an interactive installation
A generated response file contains the installation settings of a custom interactive installation. Use the response file to run a silent installation on another computer.

If you chose to create a database server during this installation, the database server is also created and configured when you run the silent installation.

To generate a custom response file:

Run the installation command with the **-r** option and specify a path and name for the response file:

- UNIX, Linux:
  1. If necessary, extract the product files by running the following command, where *filename* is the name of the product `tar` file: `tar xvf filename`
  2. Run the following command: `ids_install -r response_file_path`
- Mac OS X:
  1. Open a terminal window and go to the `/Volumes/iif.version.macosx64` directory, where *version* is the database server version number.
  2. Run the following command: `ids_install -r response_file_path`
- Windows: Run the following command from a command prompt: `ids_install.exe -i gui -r response_file_path`

Log files are created during the installation process. You can use the log files to troubleshoot installation errors.

If you created a database server during installation, the database server is configured, initialized, started, and ready for client connections. To see the connectivity information, look at the `sqlhosts` file. To see the configuration information, look at the `onconfig` file. Both files are in the `$INFORMIXDIR/etc` (UNIX, Linux, Mac OS X) or the `%INFORMIXDIR%\etc` (Windows) directory. You can customize the database server properties by setting configuration parameters in the `onconfig` file, setting environment variables in your environment, and adding connectivity information to the `sqlhosts` file. See Overview of database server configuration and administration.

After the custom response file is generated, you can use it to run a silent installation on another computer.

If you did not create a database server during installation, you must configure and initialize the database server.

Running a silent installation

You can run a silent installation command that does not require interaction.

Before you start a silent installation, complete the following tasks:

- Prepare your target systems for installation.
- Log in as the appropriate user to be the owner of the database server.
- Prepare a response file. You need different response files for UNIX-style operating systems and Windows operating systems.

On each target computer, you can customize the silent installation in the following ways:

- Accept the license. Include the **-DLICENSE_ACCEPTED=TRUE** property.
- Bypass operating system prerequisite checks. Use with caution. Include the **-DPRQCHECK=FALSE** property.
- Set the installation path. Include the **-DUSER_INSTALL_DIR=***path* property.
- Overwrite existing product installations. Use with caution. Include the **-DOVERWRITE_PRODUCT=TRUE** property.

To run a silent installation on each target computer:

1. Copy the response file and the installation media to the computer where you want to install the database server.
2. Run the silent installation command, specifying the absolute path to the response file after the `-f` option:
   - UNIX, Linux, Mac OS X: `ids_install -i silent -f` *`response_file_path`*
   - Windows: `ids_install.exe -i silent -f` *`response_file_path`*

   You can include silent installation properties in the command.
3. Complete any applicable postinstallation tasks, including initiating the disk space when you start the server for the first time. If you choose to create a database server during the installation that created the response file, the database server is configured, initialized, started, and ready for client connections.

Log files are created during the installation process. You can use the log files to troubleshoot installation errors.

Example

The following silent installation command specifies the default response file in the *media_dir* directory, specifies an installation directory of `opt/IBM/Informix` with the **-DUSER_INSTALL_DIR** property, and accepts the license with the **-DLICENSE_ACCEPTED=TRUE** property:

```
ids_install -i silent -f media_dir/bundle.properties
    -DUSER_INSTALL_DIR=opt/IBM/Informix -DLICENSE_ACCEPTED=TRUE
```

Related concepts:

Chapter 5, “Installation troubleshooting,” on page 5-1

Related tasks:

Performing a Silent Client Product Installation on UNIX and Linux (Client Products Installation Guide)

Performing a Silent Client Product Installation on Windows (Client Products Installation Guide)

Related reference:

Chapter 4, “Installation command syntax,” on page 4-1

Redistributing Informix with scripts (UNIX, Linux, Mac OS X)

You can extract IBM Informix product files from the installation media for script-based redistribution to other computers.

Complete the following prerequisite tasks:

- Prepare your systems for installation.
- Log in as **root** user to create and copy a root installation that runs with user and group **informix** accounts.

During this procedure you run the installation application to extract the installation media files, and then you redistribute the database server and client products on other computers. You can use this approach in a deployment scenario or for an embedded software solution where you want to save time and reduce the amount of required disk space.

Restriction: Do not start the database server if problems occur during the installation or you can cause further installation problems.

To install the database server with a script:

1. Run the **ids_install** installation command.
2. When you are prompted to select an option for your installation goals, select **Extract the product files (-DLEGACY option)**.
3. Follow the instructions in the installation application.
4. While logged in as the **root** user, manually install the Global Security Kit by running the **installgskit** from the `$INFORMIXDIR/gskit` directory.
5. Copy the contents of the installation directory and place them into the directory of another computer where you want to redistribute the products. The contents include the **RUNasroot** scripts that you use for redistribution.
6. Run the following script to complete the redistribution of the database server files: `RUNasroot.installserver`
7. Run the **onsecurity** utility to verify that the path where the product was redistributed is secure.

Log files are created during the installation process. You can use the log files to troubleshoot installation errors.

Complete any applicable postinstallation tasks, including initiating disk space when you start the server for the first time.

Related concepts:

Chapter 5, “Installation troubleshooting,” on page 5-1

Related tasks:

Extracting and Redistributing Client Products with Scripts (Client Products Installation Guide)

Related reference:

Chapter 4, “Installation command syntax,” on page 4-1

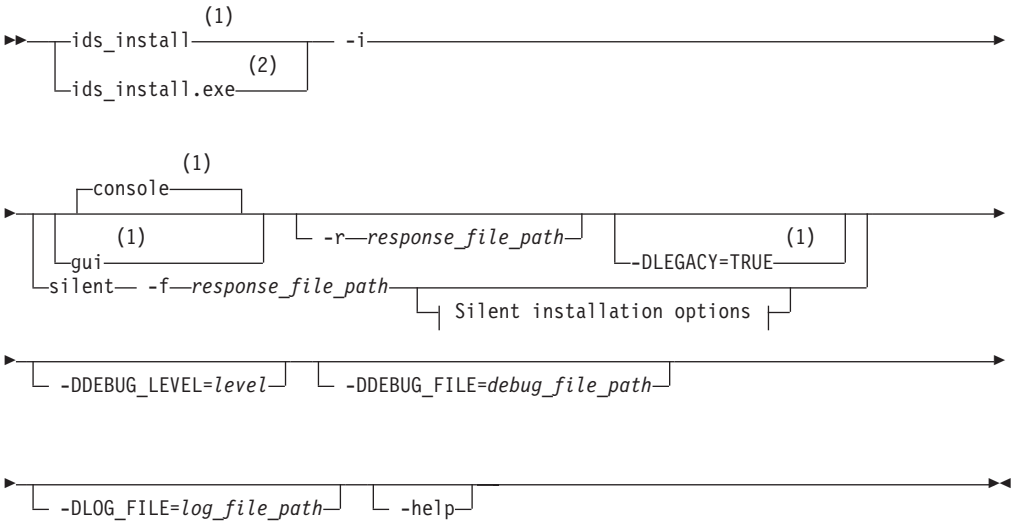
The onsecurity utility (UNIX) (Security Guide)

Chapter 4. Installation command syntax

You can include options to the installation command when you run the command from the command line.

Log in as the appropriate user to be the owner of the database server to run this command.

Syntax

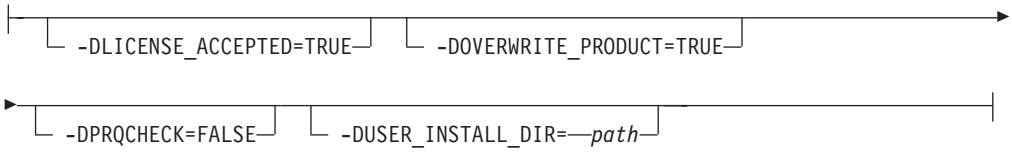


Notes:

1 UNIX, Linux, Mac OS X only

2 Windows only

Silent installation options:



Parameters

You can set properties on the installation command line that are useful for specific environments or installation methods. The values **TRUE** and **FALSE** are case-sensitive.

The following table describes the installation application options.

Table 4-1. Installation options

Option	Meaning
-DDEBUG_FILE=*debug_file_path*	Specifies the full path and name of the debug file. If tracing is enabled, the default name and location of the file is `/tmp/iad_dbg.log` (UNIX, Linux, Mac OS X) or `\tmp\iad_dbg.log` (Windows).
-DDEBUG_LEVEL=*level*	Specifies the level of tracing detail in the debug file. Possible values for *level* are 1 - 9, with 9 being the greatest level of tracing detail. By default, tracing is disabled.
-DLEGACY=TRUE	UNIX, Linux, Mac OS X: Extracts the product files from the installation media to create a script-based redistribution to other computers. For instructions, see "Redistributing Informix with scripts (UNIX, Linux, Mac OS X)" on page 3-4.
-DLICENSE_ACCEPTED=TRUE	Accepts the product license terms during a silent installation instead of indicating acceptance in the response file.
-DLOG_FILE=*log_file_path*	Specifies the full path and name of the installation log file. The default name and location of the installation log file is `/tmp/iad_act.log` (UNIX, Linux, Mac OS X) or `\tmp\iad_act.log` (Windows).
-DOVERWRITE_PRODUCT=TRUE	Specifies to overwrite an existing product installation during a silent installation. Otherwise, the installation application exits if it finds an existing product installation.
-DPRQCHECK=FALSE	Specifies to bypass the operating system prerequisites check during a silent installation. **Important:** Use with caution. Disabling the prerequisites check might result in an installation that does not function properly.
-DUSER_INSTALL_DIR=*install_path*	Specifies the installation path during a silent installation instead of specifying the path in a response file. Replace *install_path* with the absolute path to the installation directory.
-help	Displays list of supported options and their functions.
-i gui	Starts the installation program in GUI mode.
-i console	UNIX, Linux, Mac OS X: Starts the installation program in console mode.
-i silent -f *response_file_path*	Starts a silent installation that uses an existing response file. The *response_file_path* specifies the absolute path and file name for the response file. For instructions, see "Running a silent installation" on page 3-3.
-r *response_file_path*	Creates a response file that is required for a silent installation. The *response_file_path* variable specifies the absolute path and file name for the response file. For more information, see "Preparing a response file" on page 3-1). Do not name your response file `bundle.properties` or `ids.properties`.

Unsupported options

The installation user interface is based on the InstallAnywhere framework; however, the **ids_install** command does not support the following options:

- **-add**
- **-remove**
- **-repair**
- **-uninstall**

To remove your installation, use the procedures that are recommended for your operating system.

Related concepts:

Chapter 10, “Removing or modifying Informix installations,” on page 10-1

Related tasks:

“Running a silent installation” on page 3-3

“Redistributing Informix with scripts (UNIX, Linux, Mac OS X)” on page 3-4

Chapter 5. Installation troubleshooting

Installation log files and return codes can provide helpful information about a completed installation or help you identify problems encountered during an installation attempt.

Tracing

Tracing is not enabled by default. You can enable tracing, set the level of tracing detail that is logged, and set the location of the debug files. To do so, include the **-DDEBUG_LEVEL** and **-DDEBUG_FILE** options with the installation command.

Log files

When you install the database server, the installation application generates log files in the same directory as the installation media. You can change the default location of your log files and enable tracing during installation.

Installation and deployment log file

The installation and deployment log file, `iad_act.log`, is created by the installation application when you install products in the Informix software bundle.

- UNIX, Linux, Mac OS X: `/tmp/iad_act.log`
- Windows: `\tmp\iad_act.log`

Standard output and standard error log files

The stdout and stderr log files contain important information that you can use to debug installation failures.

- UNIX, Linux, Mac OS X: `/tmp/bundle_install.stdout` and `/tmp/bundle_install.stderr`
- Windows: `\tmp\bundle_install.stdout` and `\tmp\bundle_install.stderr`

Client products have separate log files.

InstallAnywhere log file

The InstallAnywhere log file contains information that is generated by the InstallAnywhere framework.

- UNIX, Linux, Mac OS X: `$INFORMIXDIR/IBM_Informix_`*`version`*`_Install_`*`date`*`.log`
- Windows: `%INFORMIXDIR%\IBM_Informix_`*`version`*`_Install_`*`date`*`.log`

If the installation application fails, the InstallAnywhere log files are placed in the following directories:

- UNIX, Linux, Mac OS X: home directory
- Windows: Desktop

InstallAnywhere return code (UNIX, Linux, Mac OS X)

You can check the InstallAnywhere return code, also known as the *exit code*, after running the installation application to learn more about the installation status and possible error conditions. To retrieve the return code, run this command after the installation application finishes:

```
echo $?
```

For information about the InstallAnywhere return code, see the documentation at the Flexera Software web site: http://www.flexerasoftware.com.

Chapter 6. Creating a database server after installation

You create a database server by setting mandatory database server properties and then starting the database server.

To create a database server:

1. Configure the mandatory properties of the database server.
   a. Set configuration parameters in the `onconfig` file.
   b. Add connectivity information in the `sqlhosts` file and other connectivity files.
   c. Set environment variables in your environment.

   Tip: On Windows operating systems, you can use the Server Instance Manager to configure the mandatory properties of the database server instead of editing the `onconfig` and `sqlhosts` file and setting environment variables.
2. Optional: Configure the wire listener for MongoDB API and REST API support.
3. Initialize disk space for the database server when you start the database server for the first time. Disk space initialization creates the initial chunk of the root dbspace, which stores control and tracking information.

   Attention: Do not initialize the disk space if you are upgrading from a previous version of the database server and you are using the same root dbspace. Initializing disk space deletes all existing user data and tracking information in the root dbspace.
   - UNIX, Linux, Mac OS X: Run the **oninit -i** command.
   - Windows: In the database server properties dialog box from the Services application, add **-i** in the **Start Parameters** field and click **Start**.

The database server is started with disk space initialized.

Windows: A program group for the server is created from which you can open a command prompt for the database server, start the Server Instance Manager, uninstall the database server, or view documentation.

After you start the database server, you can set optional configuration parameters, environment variables, and connectivity settings.

Related concepts:

Overview of database server configuration and administration (Administrator's Guide)

Install the wire listener (JSON compatibility)

Related reference:

The oninit utility (Administrator's Reference)

Initialization process (Administrator's Guide)

Setting configuration parameters

You set configuration parameters that control database server behavior in the onconfig file. The onconfig file is a text file that you create based on a template file. The onconfig file contains many configuration parameters, but most have default values and can be set while the database server is running.

Use the template configuration file, onconfig.std, as a basis for your configuration file. The onconfig.std file contains most configuration parameters set to default values. The onconfig.std file is in the $INFORMIXDIR/etc directory (UNIX, Linux, Mac OS X) or %INFORMIXDIR%\etc (Windows) directory.

Important: Do not modify or delete the onconfig.std file, which is a template and not a functional configuration.

To create an onconfig file and set the mandatory configuration parameters:

1. Copy the onconfig.std template file and save the file with a unique name, such as onconfig.*servername*, in the $INFORMIXDIR/etc directory (UNIX, Linux, Mac OS X) or %INFORMIXDIR%\etc (Windows) directory.
2. Modify your onconfig file. You can use several different methods to edit the onconfig file, including using a text editor.

   At minimum, set the following configuration parameters:

   - DBSERVERNAME: Set to the name of the database server.
   - ROOTPATH: Set to the location of the root dbspace, which contains control and tracking information for the database server.
   - SERVERNUM: Set to a unique number for the database server. If you do not have another database server on the same computer, you can leave the default value of 0.

When you set environment variables, set the **ONCONFIG** environment variable to the name of your customized configuration file.

If you omit a parameter value in your copy of the configuration file, the database server either uses default values from the onconfig.std template file or calculates values based on other parameter values.

Related concepts:

Database server configuration (Administrator's Guide)

Related tasks:

Modifying the onconfig file (Administrator's Reference)

Related reference:

onconfig Portal: Configuration parameters by functional category (Administrator's Reference)

Preparing connectivity files

Prepare the files that the database server uses to communicate with client applications and with other database servers.

The connectivity information allows a client application to connect to any IBM Informix database server on the network. The connectivity data for a particular database server includes the database server name, the type of connection that a

client can use to connect to it, the host name of the computer or node on which the database server runs, and the service name by which it is known.

You must prepare the connectivity information even if the client application and the database server are on the same computer or node. You are not required to specify all possible network connections in the `sqlhosts` file before you start the database server. However, after you add connectivity information to the `sqlhosts` file, you must restart the database server to make the new connection available.

To add connectivity information:

1. Edit the `sqlhosts` file to include the correct connectivity information with a text editor or equivalent tool.
   - The default name and location of this file is `$INFORMIXDIR/etc/sqlhosts` (UNIX, Linux, Mac OS X) or `%INFORMIXDIR%\etc\sqlhosts` (Windows).
   - If you set up several database servers to use distributed queries, use either one `sqlhosts` file to which the **INFORMIXSQLHOSTS** environment variable points or separate `sqlhosts` files in each database server directory.
2. If your system uses Internet Protocol network connections, enter settings in the following files:
   - UNIX, Linux, Mac OS X: The `/etc/hosts` and `/etc/services` files.
   - Windows: The `%windir%\system32\drivers\etc\services` file.

Related concepts:

The sqlhosts information (Administrator's Guide)

Related reference:

Connectivity files (Administrator's Guide)

Setting environment variables

Set environment variables after you install the database server. If you created a database server during installation, all mandatory environment variables are set, however, you can set optional environment variables.

You must be logged in as an administrative user or with sufficient privileges to set environment variables.

UNIX, Linux, Mac OS X: You can set environment variables at the command line, in an environment configuration file, or in a login file.

Windows: You can set environment variables in the system applet or at the command line.

To set the minimum required environment variables:

1. Set the **INFORMIXDIR** environment variable to the directory where you installed the database server.
2. UNIX, Linux, Mac OS X: Set the **PATH** environment variable to include the `$INFORMIXDIR/bin` directory as follows.
   - C shell:

     ```
     setenv PATH ${INFORMIXDIR}/bin:${PATH}
     ```

   - Bourne shell:

     ```
     PATH=$INFORMIXDIR/bin:$PATH
     export PATH
     ```

You must set the **INFORMIXDIR** variable and add `$INFORMIXDIR/bin` to the **PATH** environment variable for each user.

3. Set the **INFORMIXSERVER** environment parameter to the name of the database server.
4. Set the **ONCONFIG** environment variable to the name of a valid `onconfig` file that you created for the server.
5. If you want to use a locale or language other than the default locale of US English, set the following environment variables:
   a. Set the **CLIENT_LOCALE** environment parameter to specify a nondefault locale.
   b. Set the **DBLANG** environment parameter to specify the subdirectory of `INFORMIXDIR` that contains the customized language-specific message files that IBM Informix products use.
   c. Set the **DB_LOCALE** environment parameter.
   d. Set the **SERVER_LOCALE** environment parameter.
   e. Set the **GL_USEGLU** environment parameter if you plan to use UTF-8 character encoding. You must set the **GL_USEGLU** environment parameter before you create a database in which you plan to store UTF-8 character data.
6. Set the **INFORMIXSQLHOSTS** environment parameter to the `sqlhosts` file. The default location of this file is `$INFORMIXDIR/etc/sqlhosts` (UNIX, Linux, Mac OS X) or `%INFORMIXDIR%\etc\sqlhosts` (Windows).
7. UNIX, Linux, Mac OS X: Set the **INFORMIXTERM** environment parameter to specify whether the DB-Access utility uses the information in the `termcap` file or the `terminfo` directory. On character-based systems, the `termcap` file and `terminfo` directory determine terminal-dependent keyboard and screen capabilities, such as the operation of function keys, color, and intensity attributes in screen displays, and the definition of window borders and graphic characters.
8. UNIX, Linux, Mac OS X: If your applications are compiled with the IBM Informix Client Software Development Kit (Client SDK), set the platform-specific library path environment variable to `$INFORMIXDIR/lib` and to the product library subdirectory. See the machine notes for your client products for the appropriate environment variable name.

Related concepts:

Using environment variables on UNIX (SQL Reference)

Environment variables in Informix products (SQL Reference)

Related tasks:

Setting environment variables on Windows (SQL Reference)

Related reference:

GLS-related environment variables (GLS User's Guide)

Chapter 7. Stopping and starting the database server

Various configuration tasks require you to stop and then start the database server. Shared memory is initialized every time that the database server starts.

The database server has various operating modes.

- You can use the **onstat -** command to determine the database server mode.
- You can use the **onmode** utility to switch the database server from one mode to another.

Use the procedure for your operating system:

- "Stopping and starting the database server (UNIX, Linux, Mac OS X)"
- "Stopping and starting the database server (Windows)"

Related reference:

onmode -k, -m, -s, -u, -j: Change database server mode (Administrator's Reference)

The oninit utility (Administrator's Reference)

onstat - command: Print output header (Administrator's Reference)

Stopping and starting the database server (UNIX, Linux, Mac OS X)

You run the **onmode** utility to stop the database server, and the **oninit** utility to start the database server.

For a standard installation, log in as user **root** or user **informix**. For a non-root installation, log in as the owner of the database server.

1. To stop the database server, run the **onmode -ku** command.
2. To start the database server, run the **oninit** command. You can include options to the **oninit** command. For example, you suppress verification prompts by running the **oninit -y** command.

Stopping and starting the database server (Windows)

You stop and start the database server with the Services application.

Log in a member of the Administrators group.

1. To stop the database server, in the server properties dialog box from the Services application, click **Stop**.
2. To start the database server, in the database server properties dialog box from the Services application, type optional **oninit** options in the **Start Parameters** box, and then click **Start**. Alternatively, you can start the database server by running the **starts** command from a command prompt. You can also include options for the **oninit** command in the **starts** command.

Chapter 8. Microsoft failover clustering support (Windows)

IBM Informix database server supports Microsoft Failover Cluster (known as Microsoft Cluster Server, or MSCS, in earlier Windows releases). That built-in feature of the Windows operating system can automatically detect and respond to server or application failure, and can balance server workloads.

Important: During setup of the Microsoft failover cluster environment, stop the database server before you shut down the operating system to avoid data loss.

Refer to the Microsoft documentation for details about installation and configuration.

For a list of the Windows versions that Informix supports, go to *Informix Server System Requirements* (http://www.ibm.com/support/docview.wss?uid=swg27013343).

Chapter 9. Multiresidency

Multiple residency refers to multiple database servers and their associated shared memory and disk structures that coexist on a single computer.

When you set up multiple independent database server environments on the same computer, you can do the following tasks:

- Separate production and development environments to protect the production system from the unpredictable nature of the development environment.
- Isolate sensitive applications or databases that are critically important, either to increase security or to accommodate more frequent backups than most databases require.

  When you use multiple residency, each database server has its own configuration file. Thus, you can create a configuration file for each database server that meets its special requirements for backups, shared-memory use, and tuning priorities.
- Test distributed data transactions on a single computer. If you are developing an application for use on a network, you can use local loopback to simulate distributed data and testing on a single computer. Later, when a network is ready, you can use the application without changes to application source code.

When you plan for multiple residency on a computer, consider the following factors:

- Storage space

  Each database server must have its own dedicated storage space. You cannot use the same disk space for more than one instance of a database server. When you prepare another database server, you must repeat some of the planning that you did to install the first database server. For example, you must consider these factors:
  - UNIX, Linux, Mac OS X: Whether the files are buffered or unbuffered, and whether the files share a disk partition with another application.
  - The location of mirrored disks, if you plan to use mirroring.
  - The location of the message log.
  - The type of backups that you run.
- Shared memory

  Each database server has dedicated shared memory. Your computer must have enough shared memory for all the database servers.

Running multiple database servers on the same computer is not as efficient as running one database server. You must balance the advantages of separate database servers with the extra performance cost.

Related concepts:

Unbuffered or buffered disk access on UNIX (Administrator's Guide)

Related reference:

Local-loopback connections (Administrator's Guide)

Creating multiresident database servers (Windows)

You can set up multiple independent database server environments on the same computer.

Before you set up multiple residency, you must install one database server.

Important: You are not required to install more than one copy of the database server binary files. All instances of the same version of the database server on one computer can share the same executable files.

To set up multiple residency, use the Server Instance Manager. You can run the Server Instance Manager as a graphical program or on the command line.

To use the Server Instance Manager program, you must have administrative privileges on the database server as a member of the **Informix-Admin** group. However, local administrator privileges are sufficient, even if the database server was installed for domain use.

To create a new database server instance with the graphical Server Instance Manager program:

1. Start the Server Instance Manager from the database server program group on the **Start** > **All Programs** menu.
2. Choose an installation method, and click **Create New** to create a new instance of the database server.
3. Follow the prompts.

After you enter the required information, the Server Instance Manager installs services, records environment variables, updates the registry, and creates the `onconfig` and `sqlhost` files for the new database server instance.

Server instance manager command-line options (Windows)

The **instmgr.exe** utility is a command-line version of the Server Instance Manager that extends the ability to configure the IBM Informix database server instance with command-line options.

You must have administrative privileges on the database server. However, local administrator privileges are sufficient, even if the database server was installed for domain use.

The following table describes the **instmgr.exe** utility options.

Table 9-1. The **instmgr.exe** *utility options*

Option	Meaning
-alias *DRDA_server_alias* **-drdaport** *DRDA_port* **drdasvc** *DRDA_service_name*	Command-line for stand-alone support of DRDA® in the DBMS.
-apw *password*	Specifies the user **informix** password for a new instance.
-c -n *servername* **-apw** *informix_password*	Creates an instance of the database server.

Table 9-1. The **instmgr.exe** *utility options (continued)*

Option	Meaning
-rename -apw *informix_password* **-n** *new_name old_name*	Rename an existing server name.
-s *.ini_filename*	Initialize the specified instance in silent mode.
-f *.ini_ filename*	Retrieve some information from a partial initialization file, and the remainder from the user.
-d -n *servername*	Delete the specified instance. **Warning:** This option removes related dbspaces and deletes the data.
-dall	Delete all instances. **Warning:** This option removes related dbspaces and deletes the data.
-uall -v *version*	Upgrade all instances to the specified version. (Upgrading a single instance in a multi-instance environment is not supported.)
-r -v *version* **-n** *servername*	Revert the specified instance to the specified version.
-rall -v *version*	Revert all instances to the specified version.
-b64 *Base64_password*	Specifies Base64–encoded password.
-l	Indicates cluster installation.
-system	Create database server instance that runs as local system user instead of user **informix**. The database server must be installed as local system user to use this option.

Creating multiresident database servers (UNIX, Linux, Mac OS X)

You can set up multiple independent database server environments on the same computer.

Before you set up multiple residency, you must install one database server.

Important: You are not required to install more than one copy of the database server binary files. All instances of the same version of the database server on one computer can share binary files.

To create multiple residency of a database server:

1. Prepare a new configuration file and set the **ONCONFIG** environment variable to the new file name.
2. Set up connectivity for the new database server instance.
3. Initialize disk space for the new database server instance.
4. Prepare the backup environment for multiple residency.
5. Modify the operating system start to start the new database server instances automatically.
6. Check the **INFORMIXSERVER** environment variables for users.

Prepare a new configuration file

Each instance of the database server must have its own `onconfig` configuration file.

To set up an instance-specific `onconfig` file:

1. Make a copy of an `onconfig` file that has the basic characteristics that you want for your new database server.
2. Give the new file a name that you can easily associate with its function. For example, you might select the file name `onconfig.acct` to indicate the configuration file for a production system that contains accounting information.
3. Set the **ONCONFIG** environment variable to the file name of the new `onconfig` file. Specify only the file name, not the complete path.
4. In the new configuration file, set the following configuration parameters:

   SERVERNUM
   Specifies an integer 0 - 255 that is associated with a database server configuration. Each instance of a database server on the same host computer must have a unique SERVERNUM value.

   DBSERVERNAME
   Specifies the database server name of a database server. You can provide a useful name for the database server, such as **ifx12test** or *hostname***ifx12test**.

   MSGPATH
   Specifies a unique path and name of the message file for a database server. If multiple database servers use the same MSGPATH parameter, you cannot identify the messages from separate database server instances. For example, if you name the database server **ifx12test**, you might specify `/usr/informix/ifx12test.log` as the message log for this instance of the database server.

   ROOTPATH and ROOTOFFSET
   Used together, specify the location of the root dbspace for a database server. The root dbspace location must be unique for every database server configuration.

If you put several root dbspaces in the same partition, you can use the same value for the ROOTPATH configuration parameter. However, in that case, you must set the ROOTOFFSET configuration parameter so that the combined values of the ROOTSIZE and ROOTOFFSET configuration parameters define a unique portion of the partition.

You are not required to change the value of the ROOTNAME configuration parameter. Even if both database servers have the name **rootdbs** for their root dbspace, the dbspaces are unique because each ROOTPATH configuration parameter specify a unique location.

You can also set the MIRRORPATH and MIRROROFFSET parameters. If the root dbspace is mirrored, the location of the root dbspace mirror must be unique for each database server.

Set up TCP/IP connectivity

If you use the TCP/IP communication protocol, you might be required to add an entry to the `services` file for the new database server instance.

The sqlhosts file must have an entry for each database server. If IBM Informix products on other computers access this instance of the database server, the administrators on those computers must update their sqlhosts files.

If you plan to use Internet Protocol network connections with an instance of a database server, the system network administrator must update the hosts and services files.

Initialize disk space

Before you initialize disk space, check the setting of the **ONCONFIG** environment variable. If it is not set correctly, you might overwrite data from another database server. Initializing disk space deletes all existing user data and tracking information in the root dbspace. When you initialize disk space for a database server, the database server initializes the disk space that is specified in the current configuration file.

Important: As you create new blobspaces or dbspaces for a database server, assign each chunk to a unique location on the device. The database server prevents you from assigning more than one chunk to the same location within a single database server environment, but you must ensure that chunks that belong to different database servers do not overwrite each other.

To initialize disk space:

- UNIX, Linux, Mac OS X: Run the **oninit -i** command.
- Windows: In the database server properties dialog box from the Services application, add **-i** in the **Start Parameters** field and click **Start**.

Prepare the backup environment for multiple residency

Depending on your backup method, you must prepare the backup environment for multiple residency:

- If you use the **ON-Bar** utility, you can back up data from various database server instances to a single storage device if the storage manager allows it. The storage manager tracks what data is backed up. However, keep storage-space and logical-log backups on separate storage devices. A best practice is to keep storage-space and logical-log backups on separate storage devices.
- If you use the **ontape** utility, you must maintain separate storage space and logical log backups for each database server instance. If you can dedicate a tape drive to each database server, use the continuous logging option to back up your logical log files. Otherwise, you must plan your storage space and logical log backup schedules carefully so that use of a device for one database server instance does not cause the other database server instance to wait. You must reset the **ONCONFIG** environment variable every time that you switch backup operations from one database server instance to the other.

Modify operating system start for multiple server instances

You can ask your system administrator to modify the system start script so that each of your database server instances starts whenever the computer is rebooted; for example, after a power failure.

To start a second instance of a database server, change the **ONCONFIG** and **INFORMIXSERVER** environment variables to point to the configuration file for the

second database server and then run the **oninit** command. Do not change the values of the **INFORMIXDIR** or **PATH** environment variables.

Similarly, you can ask the system administrator to modify the shutdown script so that all instances of a database server shutdown normally.

Reset the INFORMIXSERVER environment variable

If you want the new instance of a database to be the default database server, you must reset the **INFORMIXSERVER** environment variable.

Related concepts:

Automate startup and shutdown on UNIX (Administrator's Guide)

Related reference:

SERVERNUM configuration parameter (Administrator's Reference)

DBSERVERNAME configuration parameter (Administrator's Reference)

MSGPATH configuration parameter (Administrator's Reference)

ROOTPATH configuration parameter (Administrator's Reference)

ROOTOFFSET configuration parameter (Administrator's Reference)

TCP/IP connectivity files (Administrator's Guide)

Chapter 10. Removing or modifying Informix installations

You can remove Informix products.

You can remove Informix products in the following ways:

- UNIX, Linux, Windows: You can remove the database server and all Informix client products that are installed in the same directory.
- You can remove the database server from a specific installation directory.

To modify an existing installation, remove the installation and then reinstall the product.

Related reference:

Chapter 4, “Installation command syntax,” on page 4-1

Removing or modifying installations (UNIX, Linux)

You can remove a database server and the Informix client products or remove only the database server.

You must be logged in as **root** or for non-root installations, as the owner of the database server.

If you have multiple installations of the product on the computer, the uninstallation application affects only the database server installation that is located in the `$INFORMIXDIR` path that you specify in the command. After Informix products are removed, you can manually delete the `$INFORMIXDIR` directory. It is not deleted automatically.

Removing an Informix software bundle installation

If you installed Informix client products, such asIBM Informix Client Software Development Kit (Client SDK), Informix Connect, or the IBM Informix JDBC Driver together in the same directory, you can quickly uninstall the database server and the client products at the same time.

To remove the database server and client product installations that are installed in the same folder, run one of the following commands:

- Console mode:

  ```
  $INFORMIXDIR/uninstall/uninstall_server/uninstallserver -i console
  ```

- GUI mode:

  ```
  $INFORMIXDIR/uninstall/uninstall_server/uninstallserver -i gui
  ```

Removing a database server

To remove a database server installation:

1. Run one of the following commands:

   Console mode:

   ```
   $INFORMIXDIR/uninstall/uninstall_ids/uninstallids -i console
   ```

   GUI mode:

   ```
   $INFORMIXDIR/uninstall/uninstall_ids/uninstallids -i gui
   ```

2. Follow the prompts in the uninstallation application.

Removing or modifying installations (Mac OS X)

You can remove the database server.

You must have the privilege to run the `sudo` command for uninstalling on your system.

To remove a database server installation:

1. Open a terminal window and change to the `$INFORMIXDIR` directory.
2. Make sure that the **$INFORMIXDIR** environment variable is set to the current directory.
3. Run the following command:

   ```
   sudo open uninstall/uninstall_server/uninstallserver.app
   ```

   The uninstallation application runs in the same installation mode that was used when the database server was installed.
4. Follow the prompts in the uninstallation application.

Important: If Client SDK is installed in the same directory as the database server:

- You must uninstall the database server before you uninstall Client SDK.
- Do not remove the Global Language Support and Messages features because both products have dependencies on the features.

Removing or modifying installations (Windows)

You can remove a database server and the Informix client products or remove only the database server. You can remove a database server in GUI mode or in silent mode.

You must have Windows administrator privileges to remove or modify the installation.

If you have multiple installations of the product on the computer, the uninstallation application affects only the database server installation that is in the `%INFORMIXDIR%` path that you specify in the command. After Informix products are removed, you can manually delete the `%INFORMIXDIR%` directory. It is not deleted automatically.

Removing an Informix software bundle installation

To remove the database server and all client products, start the uninstallation application by using one of the following methods:

- From the installation directory, run the `uninstallids.exe` executable file. For example, `%INFORMIXDIR%\uninstall\uninstall_ids\uninstallids.exe`.
- Select the program name in the Control Panel Programs and Features window and use the **Uninstall or Change a Program** utility to uninstall. For example, select **IBM Informix Bundle *Version***, where *Version* is the database server version number, and then click **Uninstall**.

Removing a database server

To remove the database server:

1. Start the uninstallation application by using one of the following methods:
   - Select **Start** > **Programs** > **IBM Informix** *Version* > **uninstallserver**, where *Version* is the database server version number.
   - Double-click the program name in the Control Panel Programs and Features window and run the **Uninstall or Change a Program** utility.
2. Follow the prompts in the uninstallation application.

Removing a database server in silent mode

You can create a response file by recording a GUI-based, interactive uninstallation of the database server from one location by using settings in the GUI application that you want to replicate. Then you run the silent uninstallation command, specifying the response file. A response file is not necessary if you do not have a customized installation.

To create the response file, you must have an installation that you can uninstall in a way that you plan to reuse for removal of other installations. It is not possible to do a silent uninstallation of Informix and any of the client products simultaneously.

To uninstall Informix in silent mode with a response file:

1. On a command line, start a GUI uninstallation of the database server with the `-r` option in the script:

   ```
   uninstall\uninstall_server\uninstallserver -i gui -r response_file_name
   ```

   Substitute *response_file_name* with full name and path of the response file. Append `.properties` at the end of the file name.
2. Copy the response file to the location where you want to uninstall the database server.
3. Run the silent uninstallation command, with the full path name of the response file:

   ```
   uninstall\uninstall_server\uninstallserver -i silent -f response_file_name
   ```

Appendix. Accessibility

IBM strives to provide products with usable access for everyone, regardless of age or ability.

Accessibility features for IBM Informix products

Accessibility features help a user who has a physical disability, such as restricted mobility or limited vision, to use information technology products successfully.

Accessibility features

The following list includes the major accessibility features in IBM Informix products. These features support:

- Keyboard-only operation.
- Interfaces that are commonly used by screen readers.
- The attachment of alternative input and output devices.

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

Related accessibility information

IBM is committed to making our documentation accessible to persons with disabilities. Our publications are available in HTML format so that they can be accessed with assistive technology such as screen reader software.

IBM and accessibility

For more information about the IBM commitment to accessibility, see the *IBM Accessibility Center* at http://www.ibm.com/able.

Dotted decimal syntax diagrams

The syntax diagrams in our publications are available in dotted decimal format, which is an accessible format that is available only if you are using a screen reader.

In dotted decimal format, each syntax element is written on a separate line. If two or more syntax elements are always present together (or always absent together), the elements can appear on the same line, because they can be considered as a single compound syntax element.

Each line starts with a dotted decimal number; for example, 3 or 3.1 or 3.1.1. To hear these numbers correctly, make sure that your screen reader is set to read punctuation. All syntax elements that have the same dotted decimal number (for example, all syntax elements that have the number 3.1) are mutually exclusive alternatives. If you hear the lines 3.1 USERID and 3.1 SYSTEMID, your syntax can include either USERID or SYSTEMID, but not both.

The dotted decimal numbering level denotes the level of nesting. For example, if a syntax element with dotted decimal number 3 is followed by a series of syntax elements with dotted decimal number 3.1, all the syntax elements numbered 3.1 are subordinate to the syntax element numbered 3.

Certain words and symbols are used next to the dotted decimal numbers to add information about the syntax elements. Occasionally, these words and symbols might occur at the beginning of the element itself. For ease of identification, if the word or symbol is a part of the syntax element, the word or symbol is preceded by the backslash (\) character. The * symbol can be used next to a dotted decimal number to indicate that the syntax element repeats. For example, syntax element *FILE with dotted decimal number 3 is read as 3 * FILE. Format 3* FILE indicates that syntax element FILE repeats. Format 3* * FILE indicates that syntax element * FILE repeats.

Characters such as commas, which are used to separate a string of syntax elements, are shown in the syntax just before the items they separate. These characters can appear on the same line as each item, or on a separate line with the same dotted decimal number as the relevant items. The line can also show another symbol that provides information about the syntax elements. For example, the lines 5.1*, 5.1 LASTRUN, and 5.1 DELETE mean that if you use more than one of the LASTRUN and DELETE syntax elements, the elements must be separated by a comma. If no separator is given, assume that you use a blank to separate each syntax element.

If a syntax element is preceded by the % symbol, that element is defined elsewhere. The string that follows the % symbol is the name of a syntax fragment rather than a literal. For example, the line 2.1 %OP1 refers to a separate syntax fragment OP1.

The following words and symbols are used next to the dotted decimal numbers:

? Specifies an optional syntax element. A dotted decimal number followed by the ? symbol indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element (for example, 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that syntax elements NOTIFY and UPDATE are optional; that is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.

! Specifies a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicates that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the same dotted decimal number can specify a ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the default option for the FILE keyword. In this example, if you include the FILE keyword but do not specify an option, default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP only applies to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.

***** Specifies a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be

repeated. For example, if you hear the line `5.1* data-area`, you know that you can include more than one data area or you can include none. If you hear the lines `3*, 3 HOST`, and `3 STATE`, you know that you can include `HOST`, `STATE`, both together, or nothing.

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write `HOST STATE`, but you cannot write `HOST HOST`.
3. The `*` symbol is equivalent to a loop-back line in a railroad syntax diagram.

`+` Specifies a syntax element that must be included one or more times. A dotted decimal number followed by the + symbol indicates that this syntax element must be included one or more times. For example, if you hear the line `6.1+ data-area`, you must include at least one data area. If you hear the lines `2+`, `2 HOST`, and `2 STATE`, you know that you must include `HOST`, `STATE`, or both. As for the * symbol, you can repeat a particular item if it is the only item with that dotted decimal number. The `+` symbol, like the `*` symbol, is equivalent to a loop-back line in a railroad syntax diagram.

Notices

This information was developed for products and services offered in the U.S.A. This material may be available from IBM in other languages. However, you may be required to own a copy of the product or product version in that language in order to access it.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not grant you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

For license inquiries regarding double-byte character set (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those

websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation
J46A/G4
555 Bailey Avenue
San Jose, CA 95141-1003
U.S.A.

Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

All IBM prices shown are IBM's suggested retail prices, are current and are subject to change without notice. Dealer prices may vary.

This information is for planning purposes only. The information herein is subject to change before the products described become available.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. The sample programs are provided "AS IS", without warranty of any kind. IBM shall not be liable for any damages arising out of your use of the sample programs.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

© (your company name) (year). Portions of this code are derived from IBM Corp. Sample Programs.

© Copyright IBM Corp. _enter the year or years_. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Privacy policy considerations

IBM Software products, including software as a service solutions, (“Software Offerings”) may use cookies or other technologies to collect product usage information, to help improve the end user experience, to tailor interactions with the end user, or for other purposes. In many cases no personally identifiable information is collected by the Software Offerings. Some of our Software Offerings can help enable you to collect personally identifiable information. If this Software Offering uses cookies to collect personally identifiable information, specific information about this offering’s use of cookies is set forth below.

This Software Offering does not use cookies or other technologies to collect personally identifiable information.

If the configurations deployed for this Software Offering provide you as customer the ability to collect personally identifiable information from end users via cookies and other technologies, you should seek your own legal advice about any laws applicable to such data collection, including any requirements for notice and consent.

For more information about the use of various technologies, including cookies, for these purposes, see IBM’s Privacy Policy at http://www.ibm.com/privacy and IBM’s Online Privacy Statement at http://www.ibm.com/privacy/details in the section entitled “Cookies, Web Beacons and Other Technologies”, and the “IBM Software Products and Software-as-a-Service Privacy Statement” at http://www.ibm.com/software/info/product-privacy.

Trademarks

IBM, the IBM logo, and ibm.com® are trademarks or registered trademarks of International Business Machines Corp., registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the web at "Copyright and trademark information" at http://www.ibm.com/legal/copytrade.shtml.

Adobe, the Adobe logo, and PostScript are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Itanium, and Pentium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

Microsoft, Windows, and Windows NT are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks and logos are trademarks or registered trademarks of Oracle and/or its affiliates.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, or service names may be trademarks or service marks of others.

Index

Special characters

A

B

C

D

E

F

G

H

I

J

L

M

N

O

P

R

S

T

U

V

W

Printed in USA

SC27-8018-00