

HCL Informix 14.10

Installing



Contents

- Chapter 1. Installing..... 1**
 - Informix® server..... 1
 - Installing Informix®..... 1
 - Client products..... 33
 - Informix® Client SDK..... 33
 - Open source drivers..... 57
 - DataBlade modules..... 58
 - DataBlade® Module Installation and Registration Guide..... 58
 - Informix® warehouse accelerator..... 70
 - Installation..... 70
- Index..... 77**

Chapter 1. Installing

These topics describe how to install HCL Informix® database servers, client products, and DataBlade® modules.

Installing Informix® database servers

These topics describe how to install HCL Informix® database servers.

Installing Informix®

The *Informix® Installation Guide* describes how to install, configure, and initialize the HCL Informix® database server.

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

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.

About this task

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 and configure a License Server to obtain licenses to run Informix.
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, Informix® Client Software Development Kit, and the 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 which edition that you might want to install, and whether to have the informix installer run the edition installer. If so, extract the edition installer jar file to the same folder as the informix installer.

Results

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

What to do next

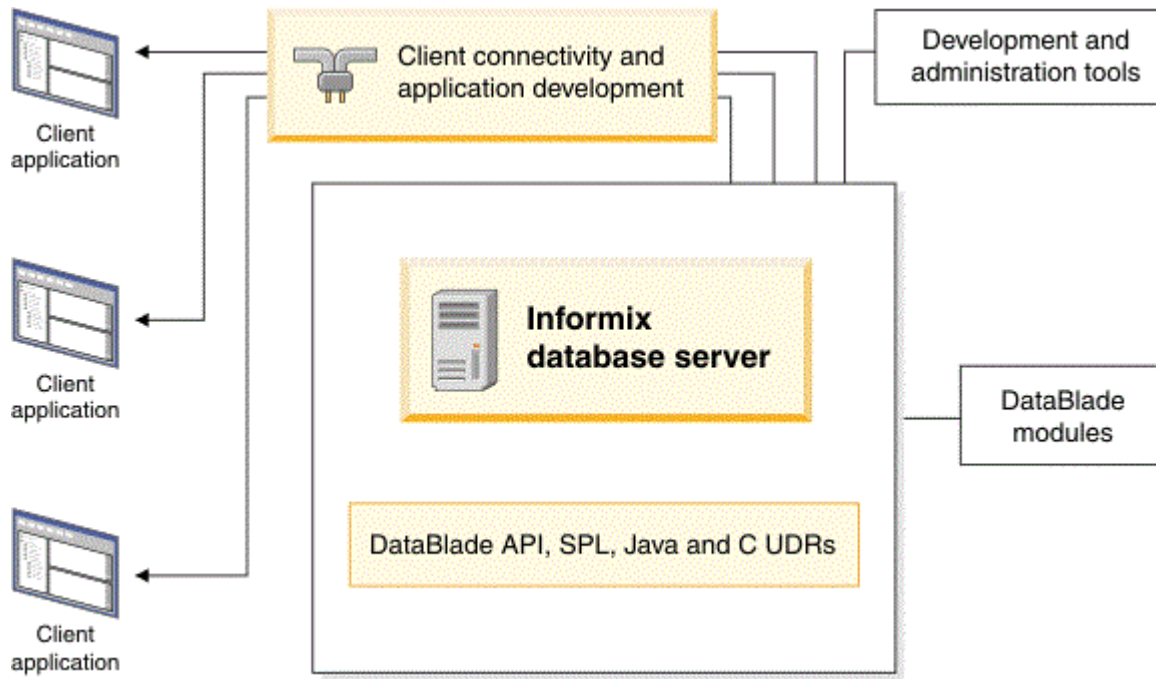
After you install the database server, you can create the demonstration databases that are included with the database server. Many examples in the documentation and in the `$(INFORMIXDIR)/demo` (UNIX, Linux, Mac OS X) or the `%INFORMIXDIR%\demo` (Windows) directory are based on the demonstration databases. 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.

Informix® components

When you install the HCL Informix® database servers, you can also install related products, such as client APIs and other products.

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

Figure 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:

Informix® Client Software Development Kit

Contains client APIs for developing and running client applications.

Informix® JDBC Driver

Java™ interfaces and classes to connect to Informix® databases.

Software Requirements

Before you install the HCL Informix® server, make sure that your computer meets these software requirements.

Operating system requirements

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

UNIX™, Linux™: 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. Turn off User Account Control (UAC) security before you install the database server.

HCL Informix® Client SDK and Server 1.0 is certified to run on 64-bit versions of these operating systems:

Table 1.

Operating System	Version
Linux x86-64	RHEL 7.4 / CentOS 7.4
	<ul style="list-style-type: none"> • Base kernel version 3.10.0 • Glibc version 2.17 • Compiler 4.8.5
	RHEL 8
	<ul style="list-style-type: none"> • Base kernel version 4.18.0 • Glibc version 2.28 • Compiler 8.2.1
	SuSE SLES 12
	<ul style="list-style-type: none"> • Minimum service pack 3 • Base kernel version 4.4.73 • Glibc version 2.22 • Compiler 4.8.5
	SuSE SLES 15
	<ul style="list-style-type: none"> • Base kernel version 4.12.14 • Glibc version 2.26 • Compiler 7.3.1
	Ubuntu 16.04 LTS
	<ul style="list-style-type: none"> • Base kernel version 4.4.0 • Glibc version 2.23 • Compiler 5.4.0
Ubuntu 18.04 LTS	
<ul style="list-style-type: none"> • Base kernel version 4.15 • Glibc version 2.27 • Compiler 7.5 	
Ubuntu 20.04 LTS	

Table 1. (continued)

Operating System	Version
	<ul style="list-style-type: none"> • Base kernel version 5.4 • Glibc version 2.31 • Compiler 9.3
AIX	7.2
Windows	Windows 2016 Windows 2019 Windows 10
Solaris Sparc	Solaris 11
Power Linux-Little Endian	RHEL 7

Application software required to run/install Server

Table 2.

Operating System	Version
Linux	OpenSSL any of the supported versions 1.0.2, 1.1.0 or 1.1.1.
AIX	OpenSSL 1.0.2 is required
Java8	JRE is required to install HCL Informix. Some components of HCL Informix also require Java8 Windows install requires a 1.8.0 JRE Linux/AIX install requires a 1.8.0 or later JRE

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** is 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

Encrypted connections and column-level encryption (CLE) are supported only if the required security-layer plug-in is installed separately on the host computer.

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.

Before you begin

You must have root authority to create users or groups.

About this task

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:

Choose from:

- AIX: `mkgroup -a n informix`
- UNIX, Linux: `groupadd -g n informix`

2. Create the user **informix** by using the `useradd` command, where *n* is a unique identifier (ID) greater than 100:

Choose from:

- `useradd -u n -g informix informix`

Result



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.

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**.

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, Informix® Client Software Development Kit, and the 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 *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 3. 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_db_sso	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.
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.

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. The spaces are not encrypted. You can encrypt the storage spaces that are created during installation by enabling storage space encryption, running a backup, and then running a restore that encrypts the spaces.

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.

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.

Before you begin



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 14](#).

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

- [Prepare your system for installation on page 1](#).
- Log in as the appropriate user to be the owner of the database server.

About this task

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.

Choose from:

- 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`

- Windows™: Double-click the `ids_install.exe` file.

2. Select the setup method, for example, **Typical** or **Custom**, and follow the instructions in the installation application.
3. Complete the installation and exit the installation application.

Results

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.

What to do next

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

Related reference

[Installation command syntax on page 16](#)

Related information

[Installation troubleshooting on page 19](#)

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.
-

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 begin

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

- [Prepare your system for installation on page 1.](#)
- Log in as the appropriate user to be the owner of the database server.

About this task

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™:
 - a. If necessary, extract the product files by running the following command, where *filename* is the name of the product tar file: `tar xvf filename`
 - b. 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`

Results

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.

What to do next

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 begin

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

- [Prepare your target systems for installation on page 1.](#)
- Log in as the appropriate user to be the owner of the database server.
- [Prepare a response file on page 14.](#) You need different response files for UNIX-style operating systems and Windows operating systems.

About this task

On each target computer, you can edit the response file to customize the silent installation in the following ways:

- Accept the license. Include the `LICENSE_ACCEPTED=TRUE` property.
- Bypass operating system prerequisite checks. Use with caution. Include the `PRQCHECK=FALSE` property.
- Set the installation path. Include the `USER_INSTALL_DIR=path` property.
- Overwrite existing product installations. Use with caution. Include the `OVERWRITE_PRODUCT=TRUE` property.
- Create and start a database server during installation. Include the `IDS_INSTANCE_CREATE=1` and `IDS_INSTANCE_INIT=1` properties.

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:

Choose from:

- 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 some of the silent installation properties in the command instead of setting them in the response file.

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 and start a database server during installation, the database server is configured, initialized, started, and ready for client connections.

Results

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

Example**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 reference

[Installation command syntax on page 16](#)

Related information

[Installation troubleshooting on page 19](#)

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**ids_install**¹**ids_install.exe**²**-i****console**¹**gui**¹**-rresponse_file_path****-DLEGACY=TRUE**¹**silent -fresponse_file_path**

Silent installation options

-DDEBUG_LEVEL=level**-DDEBUG_FILE=debug_file_path****-DLOG_FILE=log_file_path****-help**

Silent installation options

-DLICENSE_ACCEPTED=TRUE**-DOVERWRITE_PRODUCT=TRUE****-DPRQCHECK=FALSE****-DUSER_INSTALL_DIR=path****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. 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 <i>level</i> 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).

1. UNIX™, Linux™, Mac OS X only

2. Windows™ only

Table 4. Installation options (continued)

Option	Meaning
-DLICENSE_ACCEPTED=TRUE	Accepts the product license terms during a silent installation instead of indicating acceptance in the response file.
-DLOG_FILE= <i>log_file_path</i>	Specifies the full path and name of the installation log file. The default name and location of the installation log file is <code>/tmp/iad_act.log</code> (UNIX™, Linux™, Mac OS X) or <code>\tmp\iad_act.log</code> (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= <i>install_path</i>	Specifies the installation path during a silent installation instead of specifying the path in a response file. Replace <i>install_path</i> 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 <i>response_file_path</i>	Starts a silent installation that uses an existing response file. The <i>response_file_path</i> specifies the absolute path and file name for the response file. For instructions, see Running a silent installation on page 15 .
-r <i>response_file_path</i>	Creates a response file that is required for a silent installation. The <i>response_file_path</i> variable specifies the absolute path and file name for the response file. For more information, see Preparing a response file on page 14). Do not name your response file <code>bundle.properties</code> or <code>ids.properties</code> .

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 information

[Running a silent installation on page 15](#)

[Removing or modifying Informix installations on page 30](#)

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 on page 16](#).

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>.

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.

Choose from:

- 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**.

Results

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.

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.

About this task

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.
- `LICENSE_SERVER`: Set to the License Server URL used to obtain the necessary licenses to run Informix

Results

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.

Preparing connectivity files

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

About this task

The connectivity information allows a client application to connect to any 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:
Choose from:
 - UNIX, Linux, Mac OS X: The `/etc/hosts` and `/etc/services` files.
 - Windows: The `%windir%\system32\drivers\etc\services` file.
-

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.

Before you begin

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

About this task

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 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 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.

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.

About this task

- 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 \(UNIX, Linux, Mac OS X\) on page 24](#)
 - [Stopping and starting the database server \(Windows\) on page 24](#)
-

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.

Before you begin

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.

Before you begin

Log in as 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.

Microsoft failover clustering support (Windows)

HCL 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* ().

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.

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 HCL 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 5. The instmgr.exe utility options

Option	Meaning
-alias <i>DRDA_server_alias</i> -drdaport <i>DRDA_port</i> drdasvc <i>DRDA_service_name</i>	Command-line for stand-alone support of DRDA® in the DBMS.
-apw <i>password</i>	Specifies the user informix password for a new instance.
-c -n <i>servername</i> -apw <i>informix_password</i>	Creates an instance of the database server.
-rename -apw <i>informix_password</i> -n <i>new_name old_name</i>	Rename an existing server name.
-s <i>.ini_filename</i>	Initialize the specified instance in silent mode.
-f <i>.ini_filename</i>	Retrieve some information from a partial initialization file, and the remainder from the user.
-d -n <i>servername</i>	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 <i>version</i>	Upgrade all instances to the specified version. (Upgrading a single instance in a multi-instance environment is not supported.)
-r -v <i>version</i> -n <i>servername</i>	Revert the specified instance to the specified version.
-rall -v <i>version</i>	Revert all instances to the specified version.
-b64 <i>Base64_password</i>	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 **hostnameifx12test**.

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 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.

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

[Installation command syntax on page 16](#)

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 as Informix® Client Software Development Kit (Client SDK), Informix® Connect, or the 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 **HCL 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 > HCL 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 `-i` 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
```

Installing client products

These topics describe how to install HCL Informix® client products.

Installing Informix® Client SDK

This topic describes how to install Informix® Client SDK on UNIX™, Linux™, and Windows™ operating systems.

Preparing to install Client SDK

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

About this task

Preparing for installation includes reviewing system requirements, deciding on the installation owner and the installation directory, and reviewing installation options. It also includes preinstallation tasks such as obtaining the software and reviewing the machine notes.

To prepare for installation:

1. Obtain the software.
2. Review the client components that you can install.
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 client products are already installed in the directory, uninstall the existing client products.
 6. Choose an installation method.
 7. If you are installing client products on the same computer as the database server, install or upgrade the database server before you install client products.

Results

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

What to do next

After you install client products, you can configure the environment and connect to the database server.

Informix® Client Software Development Kit components

You use the Client SDK to develop and run client applications. Client SDK includes APIs, programs, and database utilities.

Applications that run on client computers require Informix® Connect to access database servers. Informix® Connect is a runtime connectivity product that is composed of runtime libraries that are included in Client SDK.

The following table describes the APIs and programs of Client SDK.

Table 6. Client SDK APIs and programs

Component	Description
Informix® .NET Provider (Windows™ only)	<p>The Informix® .NET Provider is a .NET assembly that lets .NET applications access and manipulate data in Informix® databases. It does this by implementing several interfaces in the Microsoft™ .NET Framework that are used to access data from a database. Using the Informix® .NET Provider is more efficient than accessing the HCL Informix® database through either of these methods:</p> <ul style="list-style-type: none"> • Using the Microsoft™ .NET Framework Data Provider for ODBC along with the Informix® ODBC Driver • Using the Microsoft™ .NET Framework Data Provider for OLE DB along with the Informix® ODBC Driver
IBM® Informix .NET Core Provider (Windows and Linux)	<p>The IBM® Informix .NET Core Provider is a .NET assembly that lets .NET applications access and manipulate data in Informix® databases. It does this by implementing several interfaces in .NET Core SDK/Runtime v3.1 which is shipped for Windows x64 and Linux x86_64 platforms.</p> <p>Names of all the Interfaces/Methods in IBM® Informix .NET Core Provider are same as Informix .NET Framework v4.x Provider.</p>

Table 6. Client SDK APIs and programs (continued)


Component	Description
	IBM® Informix .NET Core adds few more functionality in addition to IBM® Informix .NET Framework v4.x Provider.
Informix® ESQ/C with XA support	An SQL embedded-language product that is used to create custom C applications.
	A C++ interface to develop object-oriented client applications for use with all HCL Informix® database servers and client-side value objects for HCL Informix®
	For more information, see the Informix® Object Interface for C++ Guide on page .
(Global Language Support)	An interface that allows HCL Informix® products to use different locales that have defined conventions for a particular language, culture, or code set.
Informix® ODBC Driver with MTS support	The HCL Informix® implementation of the Open Database Connectivity (ODBC) 3.0 Level 1+ standard that supports Microsoft™ Transaction Server (MTS). This driver contains extensibility support for HCL Informix®.
	For more information, see the Informix® ODBC Driver Guide on page .
Informix® OLE DB Provider (Windows™ only)	A client-side, native OLE DB provider that implements full functionality for base-level providers and contains extensibility support for HCL Informix®.
Data Server Driver Package	The Data Server Driver Package includes drivers that are compatible with Distributed Relational Database Architecture™ (DRDA®) protocols.
	You can use the Data Server JDBC Driver to develop JDBC applications with Informix® software.
	You can use the Data Server .NET Provider to develop .NET applications with Informix® software. The Data Server .NET Provider includes support for web application development in Database Add-Ins for Visual Studio.
	Data Server Driver Package installation does not affect an existing installation of the Informix® .NET Provider or the Informix® JDBC Driver. You can use either or both of the .NET Providers or JDBC drivers.

The following table describes the HCL Informix® common database utilities in the Client SDK.

Table 7. HCL Informix® common database utilities

Component	Description
Documentation Viewer	Use the viewer to see the release and machine notes in text format.
The finderr utility on UNIX™ systems and the Informix®	Use these utilities to obtain information about error messages, including corrective actions, that are specific to HCL Informix®.

Table 7. HCL Informix® common database utilities (continued)

Component	Description
Error Messages utility on Windows™ systems	
The ILogin utility (Windows™ only)	Use the ILogin Demo utility to quickly test the connection to Informix®. Click File > Run then enter the HCL Informix® connection parameters. You must create the target database before you connect to it.
Password CSM	HCL Informix® Password Communications Support Module for client applications. Password CSM must also be installed on the Informix® database server to which the client connects.
	 Note: Support for Communication Support Module (CSM) is removed starting Informix Server 14.10.xC9 . You should use Transport Layer Security (TLS)/Secure Sockets Layer (SSL) instead.
Connection Manager	<p>The Connection Manager is a utility that can monitor the workload and status of database servers in high-availability clusters, Enterprise Replication domains, grids, and server sets.</p> <p>Connection Managers can control failover for high-availability clusters, direct client connection requests to appropriate database servers, act as proxy servers to handle client/server communication, and prioritize connections between application servers and the primary server of a high-availability cluster.</p>
DB-Access	<p>The DB-Access utility is included in the HCL Informix® Client SDK when the corresponding HCL Informix® server version is available for the same operating system. When the DB-Access utility is included in the Client SDK, it is the same utility that is delivered with the Informix® server. The utility provides a stand-alone client command-line tool that can directly access the HCL Informix® database instance. This dbaccess client supports the same user interface and functionality as the dbaccess server utility.</p>
The ConnectTest utility (Windows™ only)	<p>Use the ConnectTest utility to test the connection to the HCL Informix® server. You can start the utility from the shortcut in the Start menu or from the <code>\$INFORMIX/bin</code> directory. The utility populates the server details from the registry that is registered by the Setnet32 utility. You can manually edit the utility to connect to other servers not listed in the registry, which does not modify the registry entry. The utility shows the resulting data of the executed SQL query.</p>
Setnet32 (Windows™ only)	Use the Setnet32 utility for configuring client products. Each package might include online help, example programs, and support programs.

Software Requirements

Before you install the Client SDK, make sure that your computer meets these software requirements.

Operating system requirements

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

UNIX™, Linux™: 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. Turn off User Account Control (UAC) security before you install the database server.

HCL Informix® Client SDK and Server 1.0 is certified to run on 64-bit versions of these operating systems:

Table 8.

Operating System	Version
Linux x86-64	RHEL 7.4 / CentOS 7.4
	<ul style="list-style-type: none"> • Base kernel version 3.10.0 • Glibc version 2.17 • Compiler 4.8.5
	RHEL 8
	<ul style="list-style-type: none"> • Base kernel version 4.18.0 • Glibc version 2.28 • Compiler 8.2.1
	SuSE SLES 12
<ul style="list-style-type: none"> • Minimum service pack 3 • Base kernel version 4.4.73 • Glibc version 2.22 • Compiler 4.8.5 	
SuSE SLES 15	
<ul style="list-style-type: none"> • Base kernel version 4.12.14 • Glibc version 2.26 • Compiler 7.3.1 	
Ubuntu 16.04 LTS	

Table 8. (continued)

Operating System	Version
	<ul style="list-style-type: none"> • Base kernel version 4.4.0 • Glibc version 2.23 • Compiler 5.4.0 Ubuntu 18.04 LTS <ul style="list-style-type: none"> • Base kernel version 4.15 • Glibc version 2.27 • Compiler 7.5 Ubuntu 20.04 LTS <ul style="list-style-type: none"> • Base kernel version 5.4 • Glibc version 2.31 • Compiler 9.3
AIX	7.2
Windows	Windows 2016 Windows 2019 Windows 10

Application software required to run/install Informix ClientSDK**Table 9.**

Operating System	Version
Linux	OpenSSL any of the supported versions 1.0.2, 1.1.0 or 1.1.1.
AIX	OpenSSL 1.0.2 is required
Java8	JRE is required to install HCL Informix. Some components of Informix also require Java8 Windows install requires a 1.8.0 JRE Linux/AIX install requires a 1.8.0 or later JRE

Installation methods

The primary way to install Client SDK or Informix® Connect is with an interactive installation that prompts you to configure installation properties. After you run an interactive installation, you can run non-interactive installations that are based on the original installation.

Interactive installation

An interactive client products installation allows you to exclude unneeded products and to prepare for a silent installation by creating a response file. After you start the installation program, choose a custom installation to exclude products, or a typical installation to install all components.

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 client products installation. You cannot run a non-interactive installation on Mac OS X.

Silent installation (UNIX, Linux, Windows)

Use to install Client SDK or Informix® Connect, with the product-specific command.

By using a response file, you can replicate a specific installation configuration on multiple computers with minimal or no user interaction. Silent installation can save much time for some user scenarios.

Extraction with command-line script (UNIX, Linux)

Use to install Client SDK or Informix® Connect individually.

The extraction with command-line installation alternative can be helpful in one of the following scenarios:

- You want to install the product to redistribute it to multiple computers quickly and with minimal disk space usage
- You want to avoid using Java™ Runtime Environment (JRE) during redistribution of the product

Installation owner

The user who installs the client products is the owner of the installation.

The standard way to install client products is as the superuser **informix** with administrative privileges.

UNIX, Linux , Mac OS X

You can install client products as the superuser **informix** or as a non-root user.

User **informix** is required for root-based installations because it has the unique user identifier (UID) to manage client products, as well as the database server. The installation program automatically creates the user **informix** and the group **informix**. If you previously installed a root-based Informix® product on the computer, user and group **informix** already exist.

You can install the client products as a non-root user. Then the user account that performs the non-root installation is the owner of the installation. However, you cannot perform a non-root installation of a client product in an **\$INFORMIXDIR** directory that has a standard, root-based installation of the Informix® server.

If you are installing client products on a platform that does not support Java, you must manually create the user **informix** and the group **informix**. For instructions, see [Creating the group informix and user informix \(UNIX, Linux\) on page .](#)

Windows

You must have Administrator privileges to install client products. If you log on to Windows™ without specifying a domain, only the local system is checked. The user **informix** is created during the installation as a member of the Administrator group.

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 can contain a database server installation.
- The directory must not contain Informix® client products installations. Uninstall existing client products before you install new client products in the same directory.
- 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.
- 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 client products.



Tip: UNIX™, Linux, Mac OS X: Do not use *back* or *quit* as a string in the **INFORMIXDIR** path or in any Informix® product installation configuration settings, including passwords.

Installing client products

You can install client products as an interactive installation or an unattended installation. You can customize the installation process with the installation command.

You have the following installation method options, depending on your operating system:

- UNIX, Linux: interactive console (default), interactive GUI, silent, scripted
- Mac OS X: interactive GUI
- Windows: interactive GUI, silent

Although the most common way to run the installation program on Windows or Mac OS X operating systems is by double-clicking an executable or an app icon, you can run the installation command from the command line on all operating systems. When you run the installation command from the command line, you can include options to customize the installation process in the following ways:

- Specify the mode for the installation program: console, GUI, or silent, depending on the operating system.
- Create a response file for use in an unattended installation.
- Specify the directory for installation log files.
- Specify the level and directory for debugging information.
- Specify silent installation options.

Related reference

[Client products installation command syntax on page 45](#)

Running an interactive installation

You can install client products by running an interactive installation program. You can choose which products to install.

Before you begin

Before you start the installation program, perform the following steps:

- [Prepare your system for installation. on page 33](#)
- Log in as the appropriate user to be the owner of the client products.

About this task

You can include options to the installation command if you run the command from the command line. For example, on UNIX and Linux, you can include the `-i gui` option to run the installation program in GUI mode instead of the default console mode.

To install client products:

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 to install Client SDK: `./installclientsdk`.

Run the following command to install Informix® Connect: `./installconnect`

- 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:
 - a. If the entire media is still in `.zip` archive format, extract the media.
 - b. Exit all other applications.
 - c. Double-click the `installclientsdk.exe` file to install Client SDK, or the `installconnect.exe` file to install Informix® Connect.
2. Follow the instructions in the installation application. Choose a custom installation to exclude products, or a typical installation to install all components.
 3. Complete the installation and exit the installation application.

Results

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

You can now configure the client environment and connect to the database server.

What to do next

Windows: If you are upgrading to new release, restart your computer after the installation completes if both of the following conditions are true:

- The new installation location is different from the previous installation location
- You plan to use a distributed transaction application that uses the COM+ technology

Related reference

[Client products installation command syntax on page 45](#)

[Client installation troubleshooting on page 55](#)

[Installation methods on page 38](#)

Running a silent installation (UNIX™, Linux™, Windows)

To perform a silent installation, which is sometimes referred to as an *unattended installation*, you must create a response file that contains information about how you want the product installed. You invoke this response file in a command-line option to perform the silent installation.

Before you begin

Before you start the installation program, perform the following steps:

- [Prepare your system for installation. on page 33](#)
- Log in as the appropriate user to be the owner of the client products.

About this task

On each target computer, you can customize the silent installation command 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 deploy Client SDK or Informix® Connect in multiple directories:

1. Create a response file by doing one of the following actions:

Choose from:

- On a command line, run the appropriate installation command with the `-r` option to start an interactive installation that generates a response file.
- Edit the `csdk.properties` file or the `conn.properties` file that is on the installation media, rename the file, and configure the settings for your environment and agreement to the license terms.

2. Copy the response file and the installation media to the computer where you want to install the client products.

3. Run the silent installation command for the client products that you want to install, indicating the absolute path to the response file after the `-f` option:

UNIX, Linux:

Choose from:

- Client SDK: `./installclientsdk -i silent -f path_name`
- Informix® Connect: `./installconnect -i silent -f path_name`

Windows:

Choose from:

- Client SDK: `./installclientsdk.exe -i silent -f path_name`
- Informix® Connect: `./installconnect.exe -i silent -f path_name`

Results

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

You can now configure the client environment and connect to the database server.

What to do next

Windows: If you are upgrading to new release, restart your computer after the installation completes if both of the following conditions are true:

- The new installation location is different from the previous installation location
- You plan to use a distributed transaction application that uses the COM+ technology

Related reference

[Client products installation command syntax on page 45](#)

[Client installation troubleshooting on page 55](#)

[Installation methods on page 38](#)

Redistributing client products with scripts (UNIX, Linux)

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

Before you begin

Complete the following prerequisite tasks:

- [Prepare your systems for installation on page 33.](#)
- Log in as **root** user to create and copy a root installation that runs with user and group **informix** accounts.

About this task

Use this method if you want to redistribute the client product files without adding a Java™ runtime environment or an uninstallation application to multiple host systems. You extract the product files by starting the installation application with the `-DLEGACY=TRUE` command on one computer. After you run the installation application, copy the files in the installation directory of the first computer to redistribute the product on other computers.

To install client products with a script:

1. Run an installation command:

Choose from:

- Client SDK: `./installclientsdk -DLEGACY=TRUE`
- Informix® Connect: `./installconnect -DLEGACY=TRUE`

2. Follow the instructions in the installation application.
3. While logged in as the **root** user, manually install the Global Security Kit by running the `installgskit` from the `$(INFORMIXDIR)/gskit` directory.
4. 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.
5. Run the script for the product that you want to deploy on the other computer.
You can deploy either Client SDK or Informix® Connect in one location, not both.

Option	Description
RUNasroot.installclientsdk	Completes the redistribution of the Client SDK files.
RUNasroot.installconnect	Completes the redistribution of the Informix® Connect files.

Results

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

You can now configure the client environment and connect to the database server.

Related reference

[Client products installation command syntax on page 45](#)

[Client installation troubleshooting on page 55](#)

[Installation methods on page 38](#)

Client products 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

installclientsdk³

installclientsdk.exe⁴

installconnect³

installconnect.exe⁴

-i

console³

gui³

-r *response_file_path*

-DLEGACY=TRUE³

silent -f *response_file_path*

Silent installation options

-DDEBUG_LEVEL= *level*

-DDEBUG_FILE= *debug_file_path*

-DLOG_FILE= *log_file_path*

-help

Silent installation options

-DLICENSE_ACCEPTED=TRUE

-DOVERWRITE_PRODUCT=TRUE

-DPRQCHECK=FALSE

-DUSER_INSTALL_DIR=*path*

3. UNIX™, Linux™, Mac OS X only

4. Windows™ only

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 10. Installation options


Option	Meaning
-DDEBUG_FILE= <i>debug_file_path</i>	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= <i>level</i>	Specifies the level of tracing detail in the debug file. Possible values for <i>level</i> 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 client products with scripts (UNIX, Linux) on page 44 .
-DLOG_FILE= <i>log_file_path</i>	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™).
-DLICENSE_ACCEPTED=TRUE	Accepts the product license terms during a silent installation instead of indicating acceptance in the response file.
-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= <i>install_path</i>	Specifies the installation path during a silent installation instead of specifying the path in a response file. Replace <i>install_path</i> 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.

Table 10. Installation options (continued)

Option	Meaning
<code>-i silent -f <i>response_file_path</i></code>	<p>Starts a silent installation that uses a response file. The <i>response_file_path</i> specifies the absolute path and file name for the response file.</p> <p>For instructions, see Running a silent installation (UNIX, Linux, Windows) on page 42.</p>
<code>-r <i>response_file_path</i></code>	<p>Creates a response file from the installation. The <i>response_file_path</i> specifies the absolute path and file name for the response file. The response file contains an installation configuration that you want to deploy in more than one location.</p> <p>For more information, see Running a silent installation (UNIX, Linux, Windows) on page 42.</p> <p>Do not name your response file <code>bundle.properties</code> or <code>ids.properties</code>.</p>

Unsupported options

The installation user interface is based on the InstallAnywhere framework; however, the `installclientsdk` and `installconnect` commands do not support the following options:

- `-add`
- `-remove`
- `-repair`
- `-uninstall`

To uninstall or modify client products, use the procedures that are recommended for the client products and operating systems.

Related information

[Installing client products on page 40](#)

Post-installation tasks for client products

After you install client products, you can configure your environment and create connections to the database server.

The method for creating connections depends on the client API or program. In some cases, you edit the `sqlhosts` file and other connectivity files. For instructions, see the documentation for the specific API.

Environment variables for clients

Client SDK, Version 4.50, supports the following environment variables.

For JDBC-specific environment variables, see [Informix environment variables with the HCL Informix JDBC Driver on page](#).

The following environment variables affect all client products, unless otherwise stated.

Table 11. Environment variables for Client SDK

Environment variable	Purpose	Publication reference and possible values
CC8BITLEVEL on page	Specifies how the C-language compiler processes non-ASCII (8-bit and multibyte) characters.	Default is 2.
CLIENT_LOCALE on page	Specifies the locale that the client application uses to perform read/write operations that involve the keyboard, display, file, or printer on the client computer.	Any valid locale specifier. No default.
DB_LOCALE on page	Specifies the locale of all the databases that a client application accesses in a single connection (the database locale).	
DBANSIWARN on page	When turned on (y), causes the preprocessor to check for Informix® extensions to ANSI-standard SQL syntax at compile time and causes an application to check for Informix® extensions at run time.	Default is n.
DBCENTURY on page	Specifies how to expand values of two-digit-year DATE and DATETIME values. P = past century, R = present century, C = closest century.	Values: P , R , or C Default is R .
DBFLTMASK on page	Specifies the number of decimal digits to use when storing a DECIMAL, SMALLFLOAT, or FLOAT data type in a character buffer DB-Access only.	Values: 0-16 Default is 16.
DBLANG on page	Specifies the subdirectory of the Informix® installation directory that contains the product-specific message (.i _{em}) files.	Default is \msg.

Table 11. Environment variables for Client SDK (continued)

Environment variable	Purpose	Publication reference and possible values
DBMONEY on page	Specifies the end-user format for MONEY values so that they can conform with various international and local monetary conventions.	Default is <code>\$. , .</code> .
DBPATH on page	Identifies the database servers that contain databases that the application accesses.	
DBTEMP on page	Specifies the directory on the client computer that the client processor uses to store temporary files (also called <i>swap</i> files).	Default is <code>\tmp</code> .
DELIMIDENT on page	Indicates whether (y) or (n) to interpret strings in double quotation marks as delimited identifiers.	Default is <code>n</code> .
ESQLMF on page	Indicates whether the ESQL/C processor automatically invokes the ESQL/C multibyte filter (<i>value=1</i> and <code>CC8BITLEVEL<3</code> , or not <i>value=0</i>).	
FET_BUF_SIZE on page	Overrides the default size of the fetch buffer.	Any valid buffer size, in bytes. Default value depends on row size.
GL_DATE on page	Specifies a customized end-user format for DATE values.	
GL_DATETIME on page	Specifies a customized end-user format for DATETIME values.	
IFX_DISABLE_IPV6 on page	Disables IPv6 support for a single database instance or for a single client application.	Values: <code>yes</code> or <code>no</code> . Default is <code>no</code> .
IFX_FLAT_UCSQ on page	Overrides the global setting and directs the optimizer to use subquery flattening for all sessions.	Default is <code>0</code> .
INFORMIXCONRETRY on page	Specifies the maximum number of additional connection attempts made to a database server in the time limit that <code>CONTIME</code> specifies.	Default is <code>1</code> .
INFORMIXCONTIME on page	Specifies the number of seconds an SQL <code>CONNECT</code> statement continues to try to establish a connection before it generates an error.	Default is <code>60 (seconds)</code> .
INFORMIXDIR on page	Identifies the location of the client programs, library files, message files, header files, and other	Default is <code>C:\INFORMIX</code> .

Table 11. Environment variables for Client SDK (continued)

Environment variable	Purpose	Publication reference and possible values
	Informix® software components. INFORMIXDIR must always be set.	
INFORMIXSERVER on page	Identifies the default database server.	
INFORMIXSQLHOSTS on page	Specifies the name of the computer on which the central registry is located.	
ISL_GSK_FIPS_MODE	To disable FIPS mode on the client side.	To disable FIPS mode on the client side, set this to 'OFF'.
NODEFDAC on page	In databases not created WITH LOG MODE ANSI, setting this to <code>yes</code> prevents the PUBLIC group from receiving table privileges or routine privileges by default when tables or when owner-privileged UDRs are created.	Default is for PUBLIC to hold all table and routine privileges.
OPTMSG on page	Enables optimized message transfers (message chaining) for all SQL statements in an application.	Default is the value that LANG specifies.
OPTOFC on page	Enables optimize-OPEN-FETCH-CLOSE functionality in an ESQL/C application that uses DECLARE and OPEN statements to execute a cursor.	Default is the value that LANG specifies.
TMP	Can be used to change the location temporary space directory used during installation. This is useful if there is not enough space in the default <code>/tmp</code> directory.	Any valid path.

Configuring client products on Windows™ systems

You can use the Setnet32 utility to configure Client SDK products.

The Setnet32 utility sets or modifies environment variables and network parameters that Informix® products use at run time.

You can access the Setnet32 utility from the client products program group.

You can set the following types of information with the Setnet32 utility:

- Environment variables. The environment variables and network parameters are stored in the Windows™ system registry and are valid for every Informix® client product that you install.

To set component-specific environment variables, see individual product documentation.

- Database server network information. A client application connects to a database server that is running on a computer that can be reached through the network. To establish the connection, use Setnet32 to specify the location of the database server on the network and the network communications protocol to use. You must obtain this information from the administrator of the database server you want to use.



Requirement: If you enter a service name, it must be defined on the client computer in the `system32\drivers\etc\services` file in the Windows™ installation directory. The service definition must match the definition on the database server host computer.

- Host computer and login information. A client application can make connections only to a host computer that it can access through the network. You describe a connection to a host computer by specifying host parameters.

You can use a set of environment variables with the function call `ifx_putenv` to override the information that was entered in the **Server Information** and **Host Information** pages of the Setnet32 utility.

To use these environment variables, set the **INFORMIXSERVER** environment variable to a valid database server name. Overrides affect only that database server.

The following table lists the WIN32 override environment variables.

Table 12. WIN32 Override Environment Variables

Environment Variable	Overrides the Value For
WIN32HOST	The HOST network parameter
WIN32USER	The USER network parameter
WIN32PASS	The PASSWORD network parameter
WIN32ASKPASSATCONNECT	The PASSWORD OPTION network parameter
WIN32SERVICE	The SERVICE network parameter
WIN32PROTOCOL	The PROTOCOL network parameter

Connectivity protocols

The connectivity protocols that you can use to connect client products to the database server depend on the operating system.

The following table uses these abbreviations for protocol names:

- BSTP - Berkeley sockets using TCP/IP
- IPSP - IPC using stream pipe
- IPSM - IPC using shared memory
- IPNM - IPC using a named pipe
- SSL - Secure Sockets Layer
- TLTP - TLI using TCP/IP
- TLIX - TLI using IPX/SPX

Table 13. Connectivity protocols for client products

Platform/OS	BSTP	IPSP	IPSM	IPNM	TLTP
AIX	onsoctcp drssoctcp	onipcstr	onipcshm		
HP Itanium™ 64	onsoctcp drssoctcp	onipcstr	onipcshm		
Linux™ 32-bit	onsoctcp drssoctcp	onipcstr	onipcshm		
Linux™ AMD 64	onsoctcp drssoctcp	onipcstr	onipcshm		
Linux™ Power	onsoctcp drssoctcp	onipcstr	onipcshm		
Linux Power Little Endian	onsoctcp drssoctcp	onipcstr	onipcshm		
Linux zSeries	onsoctcp drssoctcp	onipcstr	onipcshm		
Linux ARM	onsoctcp drssoctcp	onipcstr	onipcshm		
Mac OS X	onsoctcp drssoctcp	onipcstr	onipcshm		
Solaris	onsoctcp drssoctcp	onipcstr	onipcshm		ontlitcp
Windows	onsoctcp drssoctcp			onipcnmp	

**Notes:**

- If you are using 64-bit servers: Remote 32-bit applications/tools (example: ESQL/C) can only connect to HCL Informix® using the TCP/IP protocol. Local 32-bit applications/tools can connect to the 64-bit server using either TCP/IP protocols or IPC stream pipe protocols. Local 32-bit applications/tools cannot connect to the 64-bit server using IPC shared memory protocol.
 - If you are using a NUMA system and have several IP addresses configured on the same physical interface: The HCL Informix® TCP listener thread binds to a particular IP address/port. It listens for incoming calls addressed to only one IP address. If there are multiple IP addresses configured on the system, a TLI client is not able to connect to the server. (If DNS is configured to resolve the host name on a rotary of the four IP addresses, you do not know which of those addresses has been selected by the HCL Informix® TCP listener). The solution is that in the `$INFORMIXDIR/etc/sqlhosts` file, use `"*hostname"` instead of `"hostname"`. It will make the HCL Informix® TCP listener bind to `INADDR_ANY` instead of a particular address.
 - For additional information, see the Informix® machine notes.
-

Removing client products

How you uninstall client products depends on the operating system and the installation method.

You have the following product removal methods:

- UNIX, Linux: interactive console, interactive GUI, silent, manual (only for products installed by a script)
- Mac OS X: interactive GUI
- Windows: interactive GUI

Removing client products

You can uninstall Informix® client products with an uninstallation application in GUI, console, or silent mode on UNIX™, Linux™, Mac OS X. Which mode you select depends on your objectives and system environment.

Before you begin

You must be logged in as a **root** (UNIX, Linux, Mac OS X) or an administrative user (Windows) to perform this task for root-based installations.

For non-root installations, either the installation owner or root user can remove the product. If you want to remove the product as a root user, verify that the installation is not in use or in production before uninstalling.

About this task

**Important:**



- Do not remove any Informix® products by manually deleting files.
- If Client SDK is installed in the same directory as the Informix® database server:
 - You must uninstall Informix® before you uninstall Client SDK.
 - Do not remove because both products have dependencies on this feature.

To uninstall client products:

1. From the **INFORMIXDIR** directory, run one of the following commands from the command line:

- UNIX, Linux, Mac OS X:

Client SDK: `uninstall/uninstall_csdk/uninstallcsdk`

Informix® Connect: `uninstall/uninstall_connect/uninstallconnect`

Include the `-i swing` or the `-i gui` option to uninstall in GUI mode.

- Windows:

Client SDK: `uninstall\uninstall_csdk\uninstallcsdk.exe`

Informix® Connect: `uninstall\uninstall_connect\uninstallconnect.exe`

Alternatively, you can choose the uninstallation option from the client products program group. You cannot use the **Add/Remove Programs** utility in the Control Panel to remove Informix® client products.

2. Follow the prompts to complete the uninstallation.

Results

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

Running a silent uninstallation (UNIX, Linux)

You can run a silent uninstallation of Client SDK and Informix® Connect on UNIX and Linux operating systems.

Before you begin

You must be logged in as a **root** user.

About this task

Silent uninstallation of Client SDK and Informix® Connect does not require a response file. A response file is required only if you want to customize the uninstallation. The only way to create the response file for silent uninstallation of a client product is to record a GUI or console uninstallation of the product in the first directory where you want to remove it. You cannot create the response file by modifying the template `bundle.properties` file.

To uninstall client products in silent mode:

1. **Optional:** If you want to create a response file, select a product installation that you want to uninstall with customized settings that are reusable for uninstallation of the same product on other computers.
2. **Optional:** Start a Client SDK or Informix® Connect product uninstallation application in GUI or console mode and specify the `-i -r path_name` option on the command line, replacing *path_name* with a full path to the response file that you want to record.

Example

For example, if you want to uninstall Informix® Connect in console mode and record the uninstallation in a response file at `C:/informix/uninstall.ini`, you would run:

```
$INFORMIXDIR/uninstall/uninstall_connect/uninstallconnect -i -r uninstall.ini
```

If you want to uninstall Client SDK, use the `uninstall/uninstall_csdk/uninstallclientsdk` command. The `-i -r path_name` option can be used together with the `-gui` option in the same command line if you prefer to perform the uninstallation in GUI mode.

3. **Optional:** Copy the response file to the computer where you want to uninstall the same client program.
4. Run the uninstallation command for the product with the `-i silent` option. If you are using a response file, indicate the relative or absolute path to the response file after the `-f` option.



Attention: Do not pass the `-f path_name` argument shown in the following example if you are not using a response file.

Choose from:

- Client SDK: `uninstall/uninstall_csdk/uninstallclientsdk -i silent -f path_name`
- Informix® Connect: `uninstall/uninstall_connect/uninstallconnect -i silent -f path_name`

Removing a scripted installation (UNIX™, Linux™)

If you ran the command-line script to install Client SDK, you must remove the client product files manually.

About this task

If you installed the database server with scripts on the same computer, removing client products also removes the database server. To remove the Informix® files manually, run the following command from the **INFORMIXDIR** directory as the **root** user:

```
rm -rf productfiles
```

Client 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.

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 debugging mode during installation.

Install and Deployment log file

The Install and Deployment log file `iad_act` is created for any installation by using the Informix® software bundle.

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

Standard-Out and Standard-Error log files

The Standard-Out and Standard-Error log files contain important debug information for use during 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`

InstallAnywhere log file

The InstallAnywhere log file contains information about the installation by using the InstallAnywhere framework.

- UNIX™, Linux™, Mac OS X:
 - Client SDK installation: `$(INFORMIXDIR)/IBM_Informix_Client-SDK_version_Install_date.log`
 - Informix® Connect installation: `$(INFORMIXDIR)/IBM_Informix_Connect_version_Install_date.log`
 - Client SDK installation that includes Global Language Support (GLS): `$(INFORMIXDIR)/IBM_Informix_GLS_version_Install_date.log`
- Windows™:
 - Client SDK installation: `%INFORMIXDIR%\IBM_Informix_Client-SDK_version_Install_date.log`
 - Informix® Connect installation: `%INFORMIXDIR%\IBM_Informix_Connect_version_Install_date.log`
 - Client SDK installation that includes Global Language Support (GLS): `%INFORMIXDIR%\IBM_Informix_GLS_version_Install_date.log`

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

- UNIX™, Linux™, Mac OS X: in your home directory.
- Windows™: on the Windows™ Desktop.

Debug and tracing

You can set the level of tracing detail that is used and the location of the debug file by specifying the `DDEBUG_LEVEL` and `DDEBUG_FILE` options on the installation command. By default tracing is turned off. For more information, see [Client products installation command syntax on page 45](#).

Return codes (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 the following command after the installation application finishes:

```
echo $?
```

For information about the meanings of the InstallAnywhere return codes, see the documentation at the Flexera Software website: <http://www.flexerasoftware.com>.

Open source drivers

You can use open source languages to write applications for HCL Informix® database servers.

The following languages can be used:

- PHP
- Ruby

For information about other open source languages that you can use with HCL Informix®, see www.iug.org/opensource.

PHP

HCL Informix® supports database access for client applications that are written in the PHP programming language by using a PDO (PHP Data Object) extension that functions as a database extraction layer. Two open source PDO extensions are available for Informix®: the PDO for Informix® and the PDO for Data Server clients.

PDO for Informix®

The PDO extension for Informix® is called PDO_INFORMIX. It works only with connections to Informix®. You can download PDO_INFORMIX from the PHP Extension Community Library (PECL) website at http://pecl.php.net/package/PDO_INFORMIX.

To use PDO_INFORMIX, you must install the Informix® ODBC Driver, which is part of the Informix® Client Software Development Kit (Client SDK).

Documentation to build and use PDO_INFORMIX is available from the php.net website: <http://us2.php.net/manual/en/ref.pdo-informix.php>.

PDO for Data Servers

The PDO extension for Data Servers is called PDO_IBM and is supported on Informix® 11.10, or later. PDO_IBM can be used for connections to both Informix® and DB2® data servers.

You can download PDO_IBM from the PHP Extension Community Library (PECL) website at http://pecl.php.net/package/PDO_IBM.

To use PDO_IBM, you must install the common CLI driver. The CLI driver is a component in the following clients and drivers:

- Informix® Client Software Development Kit (Client SDK)
- Data Server Client

- Data Server Runtime Client
- Data Server Driver for ODBC, CLI, and .NET (Windows™ only)
- Data Server Driver for ODBC and CLI

Documentation to build and use PDO_INFORMIX is available from php.net website: <http://us2.php.net/manual/en/ref.pdo-ibm.php>.

Ruby

HCL Informix® supports database access for client applications that are written in the Ruby programming language with two open source packages. Each package, which is known as a ruby gem, is a precompiled file that combines the Ruby driver and the Ruby on Rails adapter in a standard framework for the Ruby runtime environment.

Ruby/Informix® Gem

Ruby/Informix® is the Ruby gem for connections to Informix® database servers. You can download Ruby/Informix® from the Rubyforge website at <http://rubyforge.org/projects/ruby-informix>.

To use Ruby/Informix®, you must install the Informix® ODBC Driver, which is part of the Informix® Client Software Development Kit (Client SDK).

Ruby Gem for Data Servers

The IBM_DB gem is the Ruby gem for Data Servers. The IBM_DB gem is supported on HCL Informix® 11.10 or later. Based on the DRDA® protocol, it allows Ruby applications to access both Informix® and DB2® data servers.

You can download Ruby/Informix® from the Rubyforge website at <http://rubyforge.org/projects/rubyibm>.

To use the IBM_DB gem, you must install the common CLI driver. The CLI driver is a component in the following clients and drivers:

- Informix® Client Software Development Kit (Client SDK)
- Data Server Client
- Data Server Runtime Client
- Data Server Driver for ODBC, CLI, and .NET (Windows™ only)
- Data Server Driver for ODBC and CLI

Installing DataBlade® modules

These topics describe how to install HCL Informix® DataBlade® modules.

DataBlade® Module Installation and Registration Guide

This publication explains how to install DataBlade® modules.

This publication is for database administrators who install and register DataBlade® modules for use in a database. It is also for DataBlade® module developers who register DataBlade® modules during development.

Installing DataBlade® modules

Before you can use a DataBlade® module, make sure your database server and any clients are properly configured.

On UNIX™, you must have these environment variables set properly: **INFORMIXDIR**, **PATH**, **LD_LIBRARY_PATH**, **ONCONFIG**, and **INFORMIXSERVER**.

On Windows™, you must have these environment variables set properly: **INFORMIXDIR** and **INFORMIXSERVER**. For more information, see the *Informix® Installation Guide*.

Using a DataBlade® module in your database

About this task

To use a DataBlade® module in your database:

1. Install the DataBlade® module on Informix®.
This process is described later topics.
2. Make the DataBlade® module available to a database by registering it in that database.
 - To use an SQL function call on UNIX™ or Windows™ to register user defined DataBlades, see [Registering with the SYSBldPrepare\(\) function on page 63](#).
 - Autoregistration of standard DataBlades, see [Autoregistration of a standard DataBlade module in your database on page 63](#).

Installing your DataBlade® module on UNIX™

How you install your DataBlade® module depends on when it was released; the installation process for DataBlade® modules released after the beginning of 2007 is different from the installation process for DataBlade® modules released before 2007.

About this task

The DataBlade® module software is installed in the directory `$INFORMIXDIR/extend/datablade.version`, where *datablade* is the project name and *version* is the version number. For example, a Map DataBlade® module with a version of 4.13.UC4 would be in `$INFORMIXDIR/extend/map.4.13.UC4`.

If the DataBlade® module package you are installing has multiple DataBlade® modules, each module is unloaded into a separate directory. Each directory has its own installation script. Move to each directory using the `cd` command and run the installation script. The order of installation does not matter.



Important: After you install your DataBlade® module, read the online notes in the `/doc` directory in the DataBlade® directory.

Installing a newer DataBlade® module on UNIX™

About this task

To install a newer DataBlade® module on UNIX™:

1. Log in as the **informix** user.
2. Perform one of the following tasks depending on how you obtain the software:

Choose from:

- **CD-ROM:** Move to the CD-ROM directory.
- **Electronic download:** Follow the directions on the download site to download the product file.

3. Run the executable command.
4. Confirm the installation directory.

By default, the value of the **INFORMIXDIR** environment variable for the database server installation is selected. You can specify a different HCL Informix® instance.

Installing a pre-2007 DataBlade® module on a UNIX™ computer

About this task

To install a pre-2007 DataBlade® module on a UNIX™ computer:

1. Log in as the **informix** user.
2. Complete the following steps if you received your DataBlade® module software on a CD.

If you plan to access the product by electronic delivery, follow the instructions at the download site and then go to [step 3 on page 60](#).

- a. Move to the CD-ROM directory.

The software for each platform has its own compressed file; the platform name is included in the file name. The string *xCn* in the file name further distinguishes the product. The values for *x* and their meaning are:

F

64 bit

H

32 bit HP

U

32 bit UNIX™ or Linux™

T

32 bit Windows™

- b. Copy the compressed product file to a temporary location, such as the `/tmp` directory.
- c. Uncompress the file and restore the content with the appropriate command (such as `uncompress`, `zcat`, `tar`, `cpio`, `rpm`, `windows`).

This command creates a *datablade* directory (or directories) into which it copies the product files, where *datablade* is the DataBlade® module project name.

3. Move to the *datablade* directory, where *datablade* is the DataBlade® module project name.

4. Run the installation script: `./install`

5. Confirm the installation directory.

By default, the value of the **INFORMIXDIR** environment variable for the database server installation is selected. You can specify a different Informix® instance.

Installing your DataBlade® module on Windows™

How you install your DataBlade® module depends on when it was released; the installation process for DataBlade® modules released after the beginning of 2007 is different from the installation process for DataBlade® modules released prior to 2007.

About this task

The DataBlade® module software is installed in the directory `%INFORMIXDIR%\extend\datablade.version`, where *datablade* is the project name and *version* is the version number.



Important: After you install your DataBlade® module, read the online notes in the `\doc` directory in the DataBlade® directory.

Installing your newer DataBlade® module on a Windows™ computer

About this task

To install your newer DataBlade® module on a Windows™ computer:

1. Log in as a member of the **Informix-Admin** group.
2. Perform one of the following tasks depending on how you obtain the software:

Choose from:

- **CD-ROM:** Move to the CD-ROM directory.
- **Electronic download:** Follow the directions on the download site to download the product file.

3. Run the executable command.

4. Confirm the installation directory.

By default, the value of the **INFORMIXDIR** environment variable for the database server installation is selected. You can specify a different Informix® instance.

5. Complete the installation.

Installing your pre-2007 DataBlade® module on a Windows™ computer

About this task

To install your pre-2007 DataBlade® module on a Windows™ computer:

1. Log in as a member of the **Informix-Admin** group.
2. Load the DataBlade® module CD into your CD-ROM drive. Or, if you plan to access the product by electronic delivery, follow the instructions at the electronic site.
3. Start the Setup program in one of the following ways:

Choose from:

- In the Run dialog box, type `d:setup` and click **OK**.
- At the DOS prompt, type `d:setup` and press **ENTER**.

The *d* represents the letter of the CD-ROM drive.

4. In the installation options dialog box, select one of the following installation types:

Option	Description
Typical	Includes all components, any online help, and any examples.
Compact	Includes all components but not online help or examples.
Custom	Allows you to select which components and subcomponents you want to install.

5. Confirm the installation directory.
By default, the value of the **INFORMIXDIR** environment variable for the database server installation is selected. You can specify a different Informix® instance.
6. In the verification dialog box, make sure the destination path and selected components are correct and click **Next**.
7. Click **Finish** to exit setup.

Uninstalling your DataBlade® module on UNIX™

About this task

This topic describes how to uninstall a DataBlade® module that was released after the beginning of 2007 on a UNIX™ computer.

To uninstall a newer DataBlade® module on UNIX™:

1. Unregister the DataBlade® module. See Unregister a module.
2. Log in as the **informix** user.
3. Run the uninstall command.

Uninstalling your DataBlade® module on Windows™

About this task

This topic describes how to uninstall a DataBlade® module that was released after the beginning of 2007 on a Windows™ computer.

To uninstall a newer DataBlade® module on Windows™:

1. Unregister the DataBlade® module. See Unregistering a module.
2. Log in as the **informix** user.
3. Use Add/Remove Programs in the Control Panel or run the uninstall command.

Autoregistration of a standard DataBlade® module in a database

There is no longer a need to register any of the standard DataBlade® modules on Informix®.

The Autoregister process will detect when a datatype or access method is being used in an SQL statement, and register the blade automatically. This will take a few seconds to complete, but is a one-time operation for each database that uses the DataBlade®.

For example, creating a table with a column that is declared as a timeseries type, or inserting a row into the calendar table, will automatically register the Timeseries datablade. You will see the following information in the message log.

```
09:30:05 INFO (autoregexe 1) (EXECUTE FUNCTION sysbldprepare ('TimeSeries.*', 'create'))
09:30:05 Loading Module <$/INFORMIXDIR/extend/ifxmngr/ifxmngr.bld>
09:30:05 The C Language Module </opt/IBM/informix/extend/ifxmngr/ifxmngr.bld> loaded
```

Results



Note: The admin task scheduler must be running for this to happen automatically. This mechanism is only available for the DataBlades that are installed with the server. User defined datablades will need to use the manual SYSBldPrepare process.

Registering with the SYSBldPrepare() function

You can use the built-in SYSBldPrepare() function to register one or more DataBlade® modules or to unregister a DataBlade® module, as an alternative to using the BladeManager application. This enables you to register DataBlade® modules from any client API that supports SQL, such as DB-Access, SPL, C API, ESQL, JDBC, or ODBC, and without requiring that BladeManager be installed.

The following restrictions affect which databases can support the SYSBldPrepare() function for DataBlade® module registration:

- No cross-database support. You cannot use SYSBldPrepare() to register or unregister a DataBlade® module in any database except the local database to which your session is currently connected. To use this function in another database, you must first connect to that database and then invoke SYSBldPrepare().
- No transaction support. Do not invoke SYSBldPrepare() within transactions that you begin explicitly.

- Currently there is no support for unlogged or ANSI/ISO-compliant databases. The function is valid only in databases that support explicit transactions.
- You cannot register a DataBlade® module on a secondary server in a cluster. Register DataBlade® modules on the primary server in a cluster.

Preparing to call the SYSBldPrepare() function

About this task

This topic lists the prerequisite tasks for using the SYSBldPrepare() function to register or unregister DataBlade® modules.

To prepare to call the SYSBldPrepare() function:

1. Install and configure Informix®.

- You must have these environment variables set properly: **INFORMIXDIR** and **INFORMIXSERVER**. For more information, see the *Informix® Installation Guide*.
- Set the DB_LIBRARY_PATH configuration parameter in the `onconfig` file. The DB_LIBRARY_PATH configuration parameter specifies the location that Informix® checks for UDR or UDT shared libraries. The DB_LIBRARY_PATH configuration parameter should include `$INFORMIXDIR/extend` for DataBlade® modules. For more information, see the *Informix® Administrator's Reference*.

2. Install DataBlade® modules.

See [Installing DataBlade modules on page 59](#) for more information about how to install DataBlade® modules on UNIX™ or Windows™ systems, and how to uninstall DataBlade® modules.

3. Obtain a user identifier and its password that authorizes you to begin a user session.

If you do not have these for the Informix® instance, contact the Database Server Administrator (DBSA) and request a user ID and a password.

On Informix® instances that have set the IFX_EXTEND_ROLE configuration parameter to enable the EXTEND role, you must be granted that role (or be user **informix** or a member of the DBSA group) to be able to register or unregister DataBlade® modules in any database. If your Informix® instance has enabled the EXTEND role, but you do not hold that role, contact the DBSA and request that the EXTEND role be granted to you.

4. Connect to the database.

For information about the connect and resource access privileges that you need to connect to the database, see the *Informix® Administrator's Guide*.

5. Establish an environment in which you can issue valid SQL statements.

This can be any client API that supports SQL, such as DB-Access, SPL, C API, ESQL, JDBC, or ODBC. Refer to the documentation of your API for information about how to establish a database connection and how to call built-in functions.

Register a DataBlade® module

To register a DataBlade® module, run the `SYSBldPrepare()` function with the DataBlade® module reference and the `'create'` arguments.

For the complete calling syntax and the signature of the `SYSBldPrepare()` function, see its description in the *Informix® Guide to SQL: Syntax*.

The DataBlade® module reference is the installed name and version number of the DataBlade® module, which you can see in the `$INFORMIXDIR/extend` directory. The version string that follows the DataBlade® module name can be replaced by or truncated with the asterisk (*) wildcard, which instructs `SYSBldPrepare()` to register the highest installed version of the specified module.

The DataBlade® module reference is the installed name and version number of the DataBlade® module, which you can see in the `$INFORMIXDIR/extend` directory. The version string that follows the DataBlade® module name can be replaced by or truncated with the asterisk (*) wildcard, which instructs `SYSBldPrepare()` to register the highest installed version of the specified module.

The following call to the `SYSBldPrepare()` function instructs the database server to register the highest installed version of a DataBlade® module named `Map`:

```
EXECUTE FUNCTION sysbldprepare('map.*','create');
```

Successful invocation of the `SYSBldPrepare()` function with `'create'` as its second argument also registers any DataBlade® modules on which the module specified in the first argument is dependent. For example, suppose you create a `WorldView` DataBlade® module that is dependent on the `Map` DataBlade® module. The following SPL statement registers the specified version of the `WorldView` DataBlade® module and implicitly registers the highest installed version of the `Map` DataBlade® module on which it has a dependency:

```
LET y = sysbldprepare('worldview.1.0.TC1','create');
```

Here the `LET` statement assigns to the variable `y` the returned status code from the call to `SYSBldPrepare()`. If `y = 0`, the registration succeeded.

If the dependency `Map` DataBlade® module is already registered, no action to register it is taken, even if the currently registered version is not the highest installed version.

Calls to the `SYSBldPrepare()` function can return error messages. For information about these error messages, see [Exceptions in calls to SYSBldPrepare\(\) on page 68](#).

Register a set of DataBlade® modules

To register a list of DataBlade® modules, you can create a text file in the `$INFORMIXDIR/extend/ifxmng` directory that contains DataBlade® module references and use it as the first argument to the `SYSBldPrepare()` function.

For the complete calling syntax and the signature of the `SYSBldPrepare()` function, see its description in the *Informix® Guide to SQL: Syntax*.

Calls to the `SYSBldPrepare()` function can return error messages. For information about these error messages, see [Exceptions in calls to SYSBldPrepare\(\) on page 68](#).

Unregister a DataBlade® module

To unregister a DataBlade® module, run the SYSBldPrepare() function with the DataBlade® module reference and the `'drop'` arguments.

For the complete calling syntax and the signature of the SYSBldPrepare() function, see its description in the *Informix® Guide to SQL: Syntax*.

The following example unregisters version 4.13 of a Map DataBlade® module:

```
EXECUTE FUNCTION sysbldprepare('map.4.13','drop');
```

When you unregister a DataBlade® module with the `'drop'` argument, the asterisk notation specifies the version of the specified module that is registered in the database.

The following example unregisters the currently registered version of the Map DataBlade® module:

```
EXECUTE FUNCTION sysbldprepare('map.*','drop');
```

The `'drop'` argument does not implicitly unregister other DataBlade® modules that have dependency relationships with the module specified by the first argument. The SYSBldPrepare() function issues an error if you attempt to unregister a DataBlade® module on which another DataBlade® module that is currently registered in the database depends. For example, suppose you create a WorldView DataBlade® module that is dependent on the Map DataBlade® module. You cannot use this function to unregister the Map DataBlade® module while the WorldView DataBlade® module is still registered.

```
EXECUTE FUNCTION sysbldprepare('map.*','drop');
```

You can unregister the Map DataBlade® module without SYSBldPrepare() throwing an exception if you first unregister the dependent WorldView DataBlade® module:

```
EXECUTE FUNCTION sysbldprepare('worldview.*','drop');
EXECUTE FUNCTION sysbldprepare('map.*','drop');
```

Calls to the SYSBldPrepare() function can return error messages. For information about these error messages, see [Exceptions in calls to SYSBldPrepare\(\) on page 68](#).

Upgrade or revert a DataBlade® module

You can use the SYSBldPrepare() function to upgrade a registered DataBlade® module to a higher version, or to revert to a lower version. Like BladeManager, the SYSBldPrepare() function does not allow more than one version of the same DataBlade® module to be registered in the same database. When you register the replacement version, you do not need a second call to SYSBldPrepare() to unregister a DataBlade® module that you have replaced with another version.

Before you can change the registered version of a DataBlade® module, the other version must be installed in the `$INFORMIXDIR/extend` directory.

Upgrade to a higher version

To upgrade a DataBlade® module to a higher version, run the SYSBldPrepare() function with the DataBlade® module reference and the `'create'` arguments.

For the complete calling syntax and the signature of the SYSBldPrepare() function, see its description in the *Informix® Guide to SQL: Syntax*.

For example, suppose you create a WorldView DataBlade® module that is dependent on a Map DataBlade® module. You have installed versions 4.13.FC3 and 4.13.FC4 of the Map DataBlade® module and versions 1.00 and 2.00 of the WorldView DataBlade® module, but only the **map.4.13.FC3** and **worldview.1.00** modules are registered in the database. To upgrade from **worldview.1.00** to **worldview.2.00**, issue these function calls:

```
EXECUTE FUNCTION sysbldprepare('map.*','create');
EXECUTE FUNCTION sysbldprepare('worldview.*','create');
```

This sequence of calls has the following effects:

- The first call upgrades the Map DataBlade® module to the highest installed version.
- The second call upgrades the WorldView DataBlade® module to the highest installed version

Here the asterisk in each function call specifies the highest installed version, so no exact version string is needed in the first argument.

Revert to an earlier version

To revert a DataBlade® module to an earlier version, run the SYSBldPrepare() function with the DataBlade® module reference and the `'create'` arguments.

For the complete calling syntax and the signature of the SYSBldPrepare() function, see its description in the *Informix® Guide to SQL: Syntax*.

For example, suppose you create a WorldView DataBlade® module that is dependent on a Map DataBlade® module. You have installed versions 4.13.FC3 and 4.13.FC4 of the Map DataBlade® module and versions 1.00 and 2.00 of the WorldView DataBlade® module. The following statements revert to version 1.00 of the WorldView DataBlade® module and version 4.13.FC3 of the Map DataBlade® module:

```
EXECUTE FUNCTION sysbldprepare('worldview.1.00','create');
EXECUTE FUNCTION sysbldprepare('map.4.13.FC3','create');
```

In these function calls, you cannot use an asterisk as the version string, because 1.00 and 4.13.FC3 are not the latest installed versions of their respective modules.

View version information of SYSBldPrepare()

To determine which SYSBldPrepare() version you have been using, use the SYSBldRelease() function. This function can be useful if you contact Software Support with SYSBldPrepare() issues. You must have run the SYSBldPrepare() function at least one time before the SYSBldRelease() function can return the version string.

SYSBldRelease() is a built-in function of Informix® that takes no arguments. It returns the version string and compilation date of the SYSBldPrepare() function. The returned version string has this format:

```
major.minor.os_codeCinterim
```

Here C is a literal character, and the *major*, *minor*, *os_code*, and *interim* elements of the version string have the same semantics that these terms have in the Module Reference segment in the calling syntax of the SYSBldPrepare() function, but with no asterisk (*) wildcard notation.

The SQL statement in the following example requests the version string of the SYSBldPrepare() function for the current database:

```
EXECUTE FUNCTION sysbldrelease();
```

For the complete calling syntax and the signature of the SYSBldPrepare() function, see its description in the *Informix® Guide to SQL: Syntax*.

Troubleshooting registration problems

These topics describe problems you might experience when registering a DataBlade® module and possible solutions to the problems.

Exceptions in calls to SYSBldPrepare()

If the IFX_EXTEND_ROLE configuration parameter is set to ON, authorization to invoke the SYSBldPrepare() function is available only to the Database Server Administrator (DBSA), and others to whom the DBSA has granted the EXTEND role. By default, the DBSA is user **informix**.

The SYSBldPrepare() function issues an error if you attempt to use the `'drop'` option to unregister a DataBlade® module on which another DataBlade® module that is currently registered in the database depends.

Informix® also issues an error if SYSBldPrepare() attempts to unregister a DataBlade® module that is not registered in the database.

The following example shows an attempt to register a DataBlade® module that is not installed and the resulting error message:

```
EXECUTE FUNCTION sysbldprepare ('web.4.13', 'create');

(U0001) - registerBlade - Unable to register web.4.13
- DataBlade module not found
- check online log and sysblderrorlog table for more information
```

Runtime errors that occur can result in diagnostic error messages from SYSBldPrepare() that are not Informix® error messages. Consider the following example:

```
EXECUTE FUNCTION sysbldprepare ('ext.1.31', 'create');
```

The failure of a call to SYSBldPrepare() can return diagnostic messages like the following:

```
(U0001) - registerBlade - registration failed for ext.1.31
- required VPCLASS is missing from onconfig
- check online log and sysblderrorlog table for more information

(U0001) - registerBlade - registration failed for ext.1.31
- sbspace problem
- check online log and sysblderrorlog table for more information

(U0001) - registerBlade - registration failed for ext.1.31
```

```
- unknown error
- check online log and sysblderrorlog table for more information
```

Connection problems

If BladeManager fails to connect to a database or drops a database connection, perform the following tasks before you call Software Support:

- Check whether BladeManager connects to some databases but not to others.

You might not have sufficient permissions to work on the databases to which you cannot connect.

Informix® software can be managed most easily when a single, default user in an environment with full permissions creates databases and registers DataBlade® modules. Using a variety of permissions might cause some services to be denied, for security reasons.

- Check whether the operation that fails works correctly if a user with full permissions performs all the steps.
- Check whether BladeManager connects to databases in one GLS locale, but not in others.

If your database and client are not connected in their respective default locales, try setting the **DB_LOCALE** and **CLIENT_LOCALE** environment variables to no locale (the default) and reconnecting to the database.

- Check the database server log file for errors. You might have to ask your database administrator for the location of the server log.
- Check whether the `$INFORMIXDIR/extend` and `$INFORMIXDIR/extend/ifxmgr` directories are deleted. If these directories no longer exist, reinstall your database server.
- On UNIX™, check whether there are symbolic links to the `$INFORMIXDIR/extend` and `$INFORMIXDIR/extend/ifxmgr` directories. If there are symbolic links, reinstall your database server into a directory without symbolic links.
- Check whether the connection problem is unique to BladeManager:
 - Check whether you have resource permissions by trying to create a table. If you cannot create a table, you do not have resource permissions: have your database administrator assign you permissions.
 - On UNIX™, check whether you can connect to the same database using DB-Access. If you cannot, consult your database administrator.
 - On Windows™, check whether you can connect to the same database using a client tool, such as SQL Editor or Schema Knowledge.

If not, check your settings in Setnet32 or consult your database administrator. If you can connect with SQL Editor, but not Schema Knowledge, you might have a problem with a high-level Informix® API.

- Check the values specified for the `DB_LIBRARY_PATH` configuration parameter in the `onconfig` file. The `DB_LIBRARY_PATH` configuration parameter should include the `$INFORMIXDIR/extend` directory for DataBlade® modules.
- Check if the server is configured so that the `EXTEND` role is needed to add UDRs and UDTs. If that is the case, you must be granted the `EXTEND` role by a DBSA (by default, user **informix**).

If you had recently installed other software when you began experiencing problems with BladeManager, you might have overwritten a DLL required by BladeManager. See if reinstalling BladeManager solves the problem.

If you are still experiencing problems, contact Software Support.

Preparation failure

When BladeManager first connects to a database, it *prepares* the database for DataBlade® module registrations by creating tables and loading data from files on the server. If you receive a preparation failure error when you attempt to connect to a database with BladeManager, complete the following tasks:

- Create a new database and connect to it with BladeManager, using the same user name. If preparation does not fail, you might have a permissions problem in the original database; contact your database administrator.
- Check the BladeManager logs for the preparation log:
 - If you do not find a preparation log, check the database server log to see if a thread failed during preparation.
 - Check the preparation log for unexpected error entries. You might be able to correct some errors (for example, if the database server ran out of disk space); otherwise, consult your database administrator about the error.
- Check whether the `$INFORMIXDIR/extend` and `$INFORMIXDIR/extend/ifxmgr` directories are deleted. If these directories no longer exist, reinstall your database server.
- On UNIX™, check whether there are symbolic links to the `$INFORMIXDIR/extend` and `$INFORMIXDIR/extend/ifxmgr` directories. If there are symbolic links, reinstall your database server into a directory without symbolic links.

Registration problems

If BladeManager fails to register, unregister, or upgrade a DataBlade® module, perform the following tasks:


- Check that the permissions on the `datablade.bld` file are set to read-only. The `datablade.bld` file is the DataBlade® module shared object file located in the `$INFORMIXDIR/extend/datablade.version` directory.
- Check the log that BladeManager generated for the operation. If the log has an unexpected error entry, send the details from the log to the vendor of the DataBlade® module.
- Try to register other DataBlade® modules: for instance, the DataBlade® modules that ship with the database server. If you can register another DataBlade® module, your problem is probably specific to the DataBlade® module that failed. Read the release notes for that DataBlade® module; some modules have special requirements, such as a named sbspace.

Installing Informix® Warehouse Accelerator

These topics describe how to install HCL Informix® warehouse accelerator.

Installation

You can install on the same computer as the Informix® database server, on a separate computer, or on a cluster (including a high-availability cluster).

 **Important:** Only one instance of can be installed on a computer.

Before you install the , ensure that your computers meet the software and hardware prerequisites, and that you have decided which architecture you want to implement.

prerequisites and requirements

Ensure that your system meets the prerequisites and the requirements for . You must have certain utilities installed. Cluster installations have additional requirements.

General prerequisites

Following are the general prerequisites for :

- Only one instance of can be installed on a computer.
- Both the Informix® and versions must be at the same release and fix pack level.

System requirements

must be installed on a computer that uses a Linux™ Intel™ x86 64-bit operating system. For the detailed list of supported operating systems and hardware, see the system requirements.

Required libraries

You must have the following libraries installed on the server where is installed:

- libicu
- xerces-c

Required utilities

You must have the following utilities on the server where is installed:

- Telnet client program
- Expect utility (expect-5)
- su command

Requirements for installed on a cluster system

Following are requirements for installing on a cluster system:

- You must have a shared-disk cluster file system. For example, General Parallel File System (GPFS™).
- For maximum performance in a shared cluster environment, all cluster nodes must have the same amount of memory and processor cores.
- The user root must be able to connect to all cluster nodes by using the Secure Shell (SSH) network protocol without a password.

- If user **informix** is used for administration, user **informix** must be able to connect to all cluster nodes by using the Secure Shell (SSH) network protocol without a password.
- The file paths for the installation files must be the same on each cluster node.

Informix® Warehouse Accelerator directory structure

When you install and set up , there are several directories that are needed.

Installation directory

is installed in the directory that is specified by the INFORMIXDIR environment variable, if the variable is set in the environment in which the installer is launched. If the variable is not set, the default installation directory is `/opt/IBM/informix`. Whenever there is a reference to the file path for the *accelerator server installation directory*, the file path appears as `$IWA_INSTALL_DIR`.

Storage directory

The software resides in its own directory, referred to as the *accelerator server storage directory*. This directory stores the catalog, data marts, logs, traces, and so forth. You create this directory when you configure the accelerator server. The file path for this directory is stored in the DWADIR parameter in the `dwainst.conf` file.

Sample directory for Java™ classes

The Java™ classes that are included with the command line sample are located in the `dwa/example/cli` directory.



Tip: Information about the Java™ classes is located in the `dwa_java_reference.txt` file in the `dwa/example/cli` directory.

Preparing the Informix® database server

Before you install , configure the database server.

Before you begin

The **sysadmin** database must exist in the Informix® database server.

To configure the Informix® database server:

1. Ensure that the user **informix** has write access to the `sqlhosts` file and the directory that the file is in.
2. Define a SOCTCP network connection type in the `sqlhosts` file for the connection between the accelerator server and the database server.
3. If you do not already have a default sbspace created and configured, create the default sbspace:
 - a. In the `onconfig` file, set the SBSPACENAME configuration parameter to the name of your default sbspace.

Example

For example, to name the default sbspace `sbsp1`:

```
SBSPACENAME sbspace1 # Default sbspace name
```

You must update the `onconfig` file before you start the database server.

- b. Use the `onspaces` command to create the sbspace.

Example

The following example creates an sbspace named `sbspace1`:

```
onspaces -c -S sbspace1 -p $INFORMIXDIR/tmp/sbspace1 -o 0 -s 30000
```



Note: The size of the sbspace can be relatively small, for example 30 - 50 MB.

- c. Restart the Informix® database server.

4. **Optional:** Add a `dwavp` virtual processor. If not explicitly added, a single `dwavp` virtual processor is dynamically allocated when the first related activity occurs.

Choose from:

- To add a single `dwavp` virtual processor for the database server instance: `onmode -p +1 dwavp`.
- To permanently add a single `dwavp` virtual processor, add the following entry to the `VPCLASS` parameter in the `onconfig` file: `VPCLASS dwavp,num=1`.



Tip: In systems with significant data mart administration activity, you can define two `dwavp` virtual processors to avoid the delay of other administrative commands while loading data marts. For example, to add two `dwavp` virtual processors for the database server instance: `VPCLASS dwavp,num=2`.

Installing

You can use the graphical mode, console mode, or silent mode to install .

Before you begin

Before you begin, see the [prerequisites and requirements on page 71](#).

About this task

You can install on the same computer as your Informix® database server, on a separate computer, or on a cluster system.

You can install from the provided installation media, or you can install it after you download Informix®.

1. On the computer where you want to run the installation program, log in as user `root`.
2. From the product media or the download site, locate the HCL Informix® bundle and unpack the `iif.version.tar` file.
3. Select the installation mode that you want to use:

Choose from:

- For the graphical or console mode:
 - a. Issue the install command to start the installation program:

Installation mode	Installation command
Graphical	<code>./iwa_install -i gui</code>
Console	<code>./iwa_install -i console</code>

- b. Read the license agreement and accept the terms.
 - c. Respond to the prompts in the installation program as the program guides you through the installation.
- For the silent mode:
 - a. Make a copy of the response file template that is located in the same directory as the installation program. The name of the template is `iwa.properties`.
 - b. In the response file change the value for license from FALSE to TRUE, to indicate that you accept the license terms. For example:

```
DLICENSE_ACCEPTED=TRUE
```

- c. Issue the installation command for the silent mode. The command for the silent mode installation is:

```
./iwa_install -i silent -f "file_path"
```



Tip: Specify the absolute path for the response file.

For example, to use the silent mode with a response file called `installer.properties` that is located in the `/usr3/iwa/` directory, the command is:

```
./iwa_install -i silent -f "/usr3/iwa/installer.properties"
```

Results

- is installed in the directory that is specified by the `INFORMIXDIR` environment variable, if the `INFORMIXDIR` environment variable is set in the environment in which the installer is launched. Otherwise, the default installation directory is `/opt/IBM/informix`.
- The configuration file, `$IWA_INSTALL_DIR/dwa/etc/dwainst.conf`, is generated during the installation. This configuration file is required to start .
- The installation log file, `$IWA_INSTALL_DIR/IBM/Informix_Warehouse_Accelerator_InstallLog.log`, is generated during the installation. This log file provides information on the actions performed during installation and success or failure status of those actions.

What to do next

You must configure and start before you can use it.

Verifying the Informix® database server environment

You can use the `ondwack` script to confirm that the Informix® database server environment is set up correctly for and that the database server can connect to the accelerator server.

Before you begin




Prerequisites:

- You must be logged on as user `informix`.
- The environment of the Informix® server instance is set.

About this task

The `ondwack` script is part of the Informix® server installation and is located in the `$INFORMIXDIR/bin` directory.

The `ondwack` script verifies the following Informix® server instance prerequisites for :

- The Informix® and accelerator versions match
- The `sbspace` is added to Informix®
- The `SBSPACENAME` configuration parameter is configured correctly in the **onconfig** file
- The listen thread for `SOCTCP` is running
- The `dwapv` virtual processor is running on the standard or primary server if the scheduler is stopped
- There is a writable `sqlhosts` file and directory
- An entry for an accelerator is included in the `sqlhosts` file
-  **For secondary servers only:**
 - The `dwapv` virtual processor is running
 - The secondary server can be updated

To verify the Informix® database server environment:

Run the `ondwack` script.

Uninstalling Informix® Warehouse Accelerator

If you need to reinstall the or if you no longer want to use , you must uninstall it.

Before you begin

You must be logged on as user `root` to run the uninstaller.

About this task

To uninstall :

1. Stop the accelerator server by using the `ondwa stop` command.
2. **Optional:** To completely uninstall the files, for example in cases where you do not plan to reinstall or upgrade the product, remove the files in the shared and local directories in the accelerator server storage directory and the log files for the accelerator server node that were created by the `ondwa setup` command:
 - a. Drop all data marts from all accelerators of the accelerator server.
 - b. Remove all accelerators of the accelerator server.
 - c. Run the `ondwa clean` command.
3. Run the uninstaller program, `uninstall_iwa`, which is located in the `$INFORMIXDIR/uninstall/uninstall_iwa` directory.

Index

Special Characters

- .NET Provider 34
- /etc/hosts file
 - multiple residency 27
- /etc/services file
 - multiple residency 27
- /etc/termcap file 22
- /etc/terminfo directory 22

A

- AAO
 - see auditing analysis officer 9
- accelerator server
 - directory 72
- administrative access 5
 - group and user informix 6
- Architecture 2
- Asterisk (*) notation
 - versions of installed modules 64
 - versions of registered modules 65
- auditing analysis officer (AAO) role 9
- autoregistering 63

B

- Backups
 - and multiple residency 25
- backups, preparing for multiple residency 27
- BEGIN WORK statement 63
- BladeManager command-line interface
 - connection problems with 69
- BladeManager GUI
 - connection problems 69

C

- CC8BITLEVEL environment variable 47
- Client products
 - uninstalling 53
- Client SDK
 - Environment Variables 47
 - uninstalling 53, 54
- CLIENT_LOCALE environment variable 47
 - setting 22
- commands
 - ids_install 12
 - uninstallids 30
 - uninstalling client products 53, 54
 - uninstalling Client SDK 53, 54
 - uninstalling
 - Informix
 - Connect
 - 53, 54
 - uninstallserver 30, 31, 32
 - commandsininstallconnect
 - installclientsdk 45
 - Components 2
 - configuration file 21
 - configuration parameters 21
 - MIRROROFFSET 27
 - configuring
 - role separation 9
 - Configuring
 - dwavp 72
 - Informix
 - products
 - 50
 - sbspace 72
 - SBSpaceName configuration parameter 72
 - Configuring your server environment 63, 64

- Connecting
 - problems with 69
- Connection Manager 34
- Connections 47
 - connectivity 22
 - configuring for multiple residency 27
- Create a server
 - installation 11
- Creating
 - database server instances 26
 - creating a server 20
 - custom installation 8, 10, 12, 38

D

- Database server
 - starting 23
 - stopping 23
- database servers
 - running multiple 27
- Database servers
 - configuring 63, 64
 - creating new instances 26
 - identifying 47
- database system security officer (DBSSO)
 - role 9
- DataBlade modules
 - directory for on Windows 61
 - installing on UNIX 59
 - installing on Windows 61
 - uninstalling on UNIX 62
 - uninstalling on Windows 62
- DATE data type 47
- DATETIME data type 47
- DB_LIBRARY_PATH configuration
 - parameter 63, 64
- DB_LOCALE environment variable 47
- DBANSIWARN environment variable 47
- DBFLTMASK environment variable 47
- DBLANG environment variable
 - setting 22
- DBMONEY environment variable 47
- DBMONEYSCALE environment variable 47
- DBPATH environment variable 47
- DBSERVERNAME parameter
 - multiple residency 27
- DBSSO
 - see database system security officer 9
- DBTEMP environment variable 47
- DELIMIDENT environment variable 47
- Delimited identifiers 47
- deployment wizard 10
- directories
 - installation 7, 40
- Directories 72
 - accelerator server 72
 - documentation 72
 - for DataBlade modules on Windows 61
 - installation 72
 - samples 72
 - storage 72
- disk space
 - for multiple residency 27
- disk space requirements 3, 36
- drsoctcp 51
- dwavp
 - adding 72
 - configuring 72

- DYLD_LIBRARY_PATH environment variable,
 - setting 22

E

- environment variables
 - CLIENT_LOCALE 22
 - DBLANG 22
 - DYLD_LIBRARY_PATH 22
 - environment variables
 - LD_LIBRARY_PATH 22
 - GL_USEGLU 22
 - INFORMIXSERVER
 - setting 22
 - INFORMIXSQLHOSTS 22
 - LIBPATH 22
 - LIBPATH environment variable (AIX),
 - setting 22
 - shared library path 22
 - SHLIB_PATH 22
 - TERMCAP 22
- Environment variables
 - CC8BITLEVEL 47
 - CLIENT_LOCALE 47
 - Database locale, setting language
 - features 47
 - Database servers
 - default 47
 - DB_LOCALE 47
 - DBANSIWARN 47
 - DBFLTMASK 47
 - DBLANG environment variable 47
 - DBMONEY 47
 - DBMONEYSCALE 47
 - DBPATH 47
 - DBTEMP 47
 - DELIMIDENT 47
 - Environment variables
 - DBLANG 47
 - INFORMIXCONTIME 47
 - INFORMIXDIR 47
 - INFORMIXSERVER 47
 - OPTMSG 47
 - OPTOFC 47
 - TMP 47
 - ESQLMF 47
 - FET_BUF_SIZE 47
 - GL_DATE 47
 - GL_DATETIME 47
 - IFX_FLAT_USCQ 47
 - INFORMIXCONRETRY 47
 - INFORMIXCONTIME environment
 - variable 47
 - INFORMIXSERVER environment variable 47
 - INFORMIXSQLHOSTS 47
 - INFORMIXSQLHOSTS environment
 - variable 47
 - ISL_GSK_FIPS_MODE 47
 - MONEY data type 47
 - NCHAR data type 47
 - NVARCHAR data type 47
 - OPTMSG environment variable 47
 - OPTOFC environment variable 47
 - TMP environment variable 47
- Environment Variables
 - Client SDK 47
- ESQL/C 34
- ESQLMF environment variable 47
- exit codes 19, 55

EXTEND built-in role 64
extracting
 Global Security Kit 44
 GSKit 44
 installation media 44

F

Failover
 Microsoft cluster 24
FET_BUF_SIZE environment variable 47
files 42
 error 19
 for connectivity 22
 installation log 19
 installation tracing 19
 response 13, 14, 15, 42
 uninstallids.exe 32

G

GL_DATE environment variable 47
GL_DATETIME environment variable 47
GL_USEGLU environment variable 22
GLS: 34
group informix 5, 6
groupadd utility 6, 6

H

hosts file 22

I

ids_install command 12, 16
IFX_EXTEND_ROLE configuration parameter 64
IFX_FLAT_USCQ environment variable 47
ILogin utility 34
Informix
 Connect
 definition 34
 uninstalling 53, 54
 informix group 5
 informix user 5
 Informix
 Warehouse Accelerator
 installing 73
 verify setup 74
 Informix-Admin group 9, 61
 using Server Instance Manager 26
 INFORMIXCONRETRY environment variable 47
 INFORMIXDIR 7, 40
 lib directory 22
 INFORMIXDIR environment variable 64
 INFORMIXSERVER environment variable 64
 setting 22
 INFORMIXSQLHOSTS environment variable,
 setting 22
 installation
 commands 16
 custom 10, 12
 debugging 19
 directory 7, 40
 disk space 3, 36
 error 19
 log files 19
 silent 13, 14, 15, 42
 typical 12
 typical or custom 8, 38
 unattended 13, 42
 Installation
 create a server 11
 preparing for 1, 33
 Installation script for DataBlade modules on
 UNIX 60

installclientsdk command 45
installconnect command 45
installing 70
 by extracting media with script 44
Installing
 DataBlade modules on UNIX 59
 DataBlade modules on Windows 61
 directory 72
 Informix
 Warehouse Accelerator
 73
 overview 70
 verify server setup 74
interactive installation 8, 38

J

Java runtime environment requirements 3, 36
JRE requirements 3, 36

L

LD_LIBRARY_PATH environment variable,
setting 22
legacy installation support 44
log files
 deploy 19, 55
 error 19, 55
 iad_act 55
 iad_act.log 19
 install 19, 55
 output 19, 55

M

Memory requirements, and multiple
residency 25
Message file 47
Message log, for multiple residency 25
Microsoft Cluster Server 24
Microsoft Failover Server 24
Mirroring, and multiple residency 25
MIRROROFFSET configuration parameter
 multiple residency 27
MIRRORPATH parameter
 multiple residency 27
MSGPATH parameter
 multiple residency 27
Multiple residency 25
 and backups 25
 and mirroring 25
 and multiple binary files, warning 26
 memory requirements 25
 message log for 25
 onconfig file 26
 planning for 25
 storage-space requirements 25
multiresident database servers 27

N

non-interactive installation 8, 38
non-root install 5
non-root installation 7, 39, 40

O

Object Interface for C++ 34
ODBC-MTS 34
OLE DB Provider 34, 34
onconfig file, and multiple residency 26
onconfig parameters 21
onconfig.std file 21
ondwackh script 74
oninit command 24
onipcnmp 51
onipcshh 51

onipcstr 51
onmode -s command 24
onsecurity utility 7
onsocimc 51
onsoctcp 51
ontliimc 51
ontlitcp 51
operating system 71
operating system requirements 3, 36
Operating systems
 supported 71

P

passwd utility 6
Password CSM 34
Planning
 installation 1, 33
Preparation
 problems with 70
Preparing
 for installation 1, 33
Prerequisites
 operating system 71
 software 71
Privileges
 local 26
 Server Instance Manager 26
Processor, storing temporary files 47
Products 2

R

redistributing
 Informix
 products
 44
Registering DataBlade modules
 problems with 70
reinstalling 75
Reinstalling 75
removing features 30
response file 13, 14, 15, 42
return codes 19, 55
role separation
 configuring 9
roles 9
ROOTOFFSET configuration parameter
 multiple residency 27
ROOTPATH parameter
 multiple residency 27

S

sbspace
 configuring 72
SBSPACENAME configuration parameter 72
scripts
 for installing by extracting media 44
server configuration 20
server instance
 creating during installation 12
 non-root installation 12
Server Instance Manager
 command-line options 26
 DRDA (Distributed Relational Database
 Architecture) 26
 Informix-Admin Group 26
 privileges 26
 starting 26
SERVERNUM configuration parameter
 multiple residency 27
Services application 24
services file 22
Setnet32 utility

- product configuration 50
- Setup program for DataBlade modules on Windows 61
- shared-library path environment variable setting 22
- SHLIB_PATH environment variable (HP-UX) setting 22
- shutdown script
 - multiple residency 27
- silent installation 13, 14, 15, 42
- sqlhosts file 22
 - multiple residency 27
- Starting the server 24, 24
- starts command 24
- startup script
 - multiple residency 27
- Stopping the server 24, 24
- Storage-space requirements, and multiple residency 25
- SYSBldPrepare built-in function 63
- SYSBldPrepare Function:
 - Registering an installed module 64
 - Unregistering a registered module 65
- SYSBldPrepare()
 - error messages 68
 - problems with 68
- SYSBldRelease built-in function 67
- system startup script
 - multiple residency 27

T

- TERMCAP environment variable setting 22
- Troubleshooting BladeManager 68, 70
- typical installation 8, 12, 38

U

- unattended installation 13
 - see also silent installation 42
- Uninstall
 - client products 53
- uninstallids command 30
- uninstallids.exe file 32
- uninstalling 30, 30, 31, 32, 75
 - database server 30, 31, 32
 - Informix
 - software bundle
 - 30, 31, 32
- Uninstalling 75
 - DataBlade modules on UNIX 62
 - DataBlade modules on Windows 62
- Uninstalling client products
 - silent 54
- uninstallserver command 30, 31, 32
- UNIX
 - installing DataBlade modules on 59
 - uninstalling DataBlade modules on 62
- Unregistering DataBlade modules
 - problems with 70
- Upgrading DataBlade modules
 - problems with 70
- user defined DataBlades 63
- user informix 6
- User informix 5
- useradd utility 6
- utilities
 - groupadd 6
 - passwd 6
 - useradd 6

V

- Version string

- of an installed DataBlade module 64
- of SYSBldPrepare 67

W

- Windows
 - Informix
 - Admin group in
 - 61
 - installing DataBlade modules on 61
 - uninstalling DataBlade modules on 62
 - wire listener 20