Informix Product Family
Informix DataBlade Developers Kit
Version 4.20

IBM Informix DataBlade Module Installation and Registration Guide

GC27-4517-00

Informix Product Family
Informix DataBlade Developers Kit
Version 4.20

IBM Informix DataBlade Module Installation and Registration Guide

GC27-4517-00

Note
Before using this information and the product it supports, read the information in "Notices" on page C-1.

This document contains proprietary information of IBM. It is provided under a license agreement and is protected by copyright law. The information contained in this publication does not include any product warranties, and any statements provided in this manual should not be interpreted as such.

When you send information to IBM, you grant IBM a nonexclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

© Copyright IBM Corporation 1996, 2013.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.

Contents

© Copyright IBM Corp. 1996, 2013

Introduction

About this publication

This publication explains how to install DataBlade® modules and how to use BladeManager to manage them in IBM® Informix® databases. BladeManager is an application that runs on client computers and requires a connection to IBM Informix.

Types of users

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.

Hardware and software requirements

BladeManager requires IBM Informix and the IBM Informix Client Software Development Kit (Client SDK). See the BladeManager release notes for version compatibility and system requirements.

BladeManager is available as both a graphical user interface and a command-line interface.

The BladeManager graphical user interface runs on personal computers with Intel processors running the Windows operating systems.

The BladeManager command-line interface runs on Windows, UNIX, and Linux computers.

Example code conventions

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

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

```
CONNECT TO stores_demo
...

DELETE FROM customer
   WHERE customer_num = 121
...

COMMIT WORK
DISCONNECT CURRENT
```

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

© Copyright IBM Corp. 1996, 2013

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

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

Additional documentation

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

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

Compliance with industry standards

IBM Informix products are compliant with various standards.

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

The IBM Informix Geodetic DataBlade Module supports a subset of the data types from the *Spatial Data Transfer Standard (SDTS)—Federal Information Processing Standard 173*, as referenced by the document *Content Standard for Geospatial Metadata*, Federal Geographic Data Committee, June 8, 1994 (FGDC Metadata Standard).

How to provide documentation feedback

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

Use one of the following methods:

- Send email to docinf@us.ibm.com.
- In the Informix information center, which is available online at http://www.ibm.com/software/data/sw-library/, open the topic that you want to comment on. Click the feedback link at the bottom of the page, fill out the form, and submit your feedback.
- Add comments to topics directly in the information center and read comments that were added by other users. Share information about the product documentation, participate in discussions with other users, rate topics, and more!

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

We appreciate your suggestions.

Chapter 1. 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**. For more information, see the *IBM Informix Administrator's Guide*.

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

Using a DataBlade module in your database

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 with the BladeManager utility, or with an SQL function call.
   - To use an SQL function call on UNIX or Windows, see Chapter 2, "Registering with the SYSBldPrepare( ) function," on page 2-1.
   - To use a graphical user interface on Windows, see Chapter 3, "Registering with the BladeManager graphical user interface," on page 3-1.
   - To use a command-line interface on UNIX or Windows, see Chapter 4, "Registering with the BladeManager command-line interface," on page 4-1.

For more information about your DataBlade module, see the user's guide for the module.

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.

The DataBlade module software is installed in the directory `$INFORMIXDIR/extend/`*`datablade.version`*, where *datablade* is the project name and *version* is the version number. For example, the IBM Informix Geodetic DataBlade Module, Version 3.12.UC1, is in `$INFORMIXDIR/extend/geodetic.3.12.UC1`.

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

To install a newer DataBlade module on UNIX:

1. Log in as the **informix** user.

© Copyright IBM Corp. 1996, 2013

2. Perform one of the following tasks depending on how you obtain the software:
   - **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. See the *Quick Start Guide* for your DataBlade module for details on this 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 IBM Informix instance.

Installing a pre-2007 DataBlade module on a UNIX computer

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.
   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 *x*C*n* 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**, **winzip**).

      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 IBM 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.

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, the IBM Informix Geodetic DataBlade Module, Version 3.12.TC1, is in $INFORMIXDIR/extend/geodetic.3.12.TC1..

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

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:
   - **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. See the *Quick Start Guide* for your DataBlade module for details on this 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 IBM Informix instance.
5. Complete the installation.

Installing your pre-2007 DataBlade module on a Windows computer

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:
   - 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 IBM 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

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 DataBlade module" on page 4-6.
2. Log in as the **informix** user.
3. Run the uninstall command. See the *Quick Start Guide* for your DataBlade module for details on this command.

Uninstalling your DataBlade module on Windows

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 DataBlade module" on page 3-4.
2. Log in as the **informix** user.
3. Use Add/Remove Programs in the Control Panel or run the uninstall command. See the *Quick Start Guide* for your DataBlade module for details on this command.

Chapter 2. 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

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 IBM Informix.
   - You must have these environment variables set properly: **INFORMIXDIR** and **INFORMIXSERVER**. For more information, see the *IBM Informix Installation Guide for UNIX, Linux, and Mac OS X* or the *IBM Informix Installation Guide for Windows*.
   - 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 *IBM Informix Administrator's Reference*.
2. Install DataBlade modules. See Chapter 1, "Installing DataBlade modules," on page 1-1 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

© Copyright IBM Corp. 1996, 2013

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 *IBM 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 *IBM 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 following call to the **SYSBldPrepare( )** function instructs the database server to register the highest installed version of the IBM Informix Geodetic DataBlade Module:

```
EXECUTE FUNCTION sysbldprepare('geodetic.*','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 IBM Informix Geodetic DataBlade Module. The following SPL statement registers the specified version of the WorldView DataBlade module and implicitly registers the highest installed version of the IBM Informix Geodetic 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 IBM Informix Geodetic 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 A-1.

Register a set of DataBlade modules

To register a list of DataBlade modules, you can create a text file in the `$INFORMIXDIR/extend/ifxmngr` 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 *IBM 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 A-1.

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 *IBM Informix Guide to SQL: Syntax*.

The following example unregisters version 2.0 of the IBM Informix Geodetic DataBlade Module:

```
EXECUTE FUNCTION sysbldprepare('geodetic.3.12','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 IBM Informix Geodetic DataBlade Module:

```
EXECUTE FUNCTION sysbldprepare('geodetic.*','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 IBM Informix Geodetic DataBlade Module. You cannot use this function to unregister the IBM Informix Geodetic DataBlade Module while the WorldView DataBlade module is still registered.

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

You can unregister the IBM Informix Geodetic DataBlade Module without **SYSBldPrepare( )** throwing an exception if you first unregister the dependent WorldView DataBlade module:

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

Calls to the **SYSBldPrepare( )** function can return error messages. For information about these error messages, see "Exceptions in calls to SYSBldPrepare( )" on page A-1.

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 *IBM Informix Guide to SQL: Syntax*.

For example, suppose you create a WorldView DataBlade module that is dependent on the IBM Informix Geodetic DataBlade Module. You have installed versions 3.12.FC1 and 3.12.FC2 of the IBM Informix Geodetic DataBlade Module and versions 1.00 and 2.00 of the WorldView DataBlade module, but only the **geodetic.3.12.FC1** 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('geodetic.*','create');
EXECUTE FUNCTION sysbldprepare('worldview.*','create');
```

This sequence of calls has the following effects:

- The first call upgrades the IBM Informix Geodetic 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 *IBM Informix Guide to SQL: Syntax*.

For example, suppose you create a WorldView DataBlade module that is dependent on the IBM Informix Geodetic DataBlade Module. You have installed versions 3.12.FC1 and 3.12.FC2 of the IBM Informix Geodetic 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 3.12.FC1 of the IBM Informix Geodetic DataBlade Module:

```
EXECUTE FUNCTION sysbldprepare('worldview.1.00','create');
EXECUTE FUNCTION sysbldprepare('geodetic.3.12.FC1','create');
```

In these function calls, you cannot use an asterisk as the version string, because 1.00 and 3.12.FC1 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 IBM 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 IBM 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 *IBM Informix Guide to SQL: Syntax*.

Chapter 3. Registering with the BladeManager graphical user interface

To use a DataBlade module in a database, you must first register the DataBlade module in the database. *Registration* is the process of executing the SQL statements that create the DataBlade module database objects and identify the DataBlade module shared object file or dynamic link library to the database server. You use BladeManager to register DataBlade modules.

Before you can use BladeManager, you must complete the tasks listed in the section "Preparing to use the BladeManager graphical user interface."

These topics describe how to register DataBlade modules using the BladeManager graphical user interface main application window on Windows computers. You can also use the **SYSBldPrepare( )** function or the command-line interface of BladeManager to register DataBlade modules. For details of DataBlade module registration on UNIX or Windows using calls to **SYSBldPrepare( )**, see Chapter 2, "Registering with the SYSBldPrepare( ) function," on page 2-1. For details of DataBlade module registration on UNIX or Windows using the command-line interface of BladeManager, see Chapter 4, "Registering with the BladeManager command-line interface," on page 4-1.

The graphical user interface to the BladeManager application provides four tabbed pages on which you can perform tasks.

Preparing to use the BladeManager graphical user interface

This topic lists the prerequisite tasks for using BladeManager to register DataBlade modules.

To prepare to use BladeManager:

1. Install and configure IBM Informix.
   - You must have these environment variables set properly: **INFORMIXDIR** and **INFORMIXSERVER**. For more information, see the *IBM Informix Installation Guide for Windows*.
   - 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 *IBM Informix Administrator's Reference*.
2. Install DataBlade modules. See Chapter 1, "Installing DataBlade modules," on page 1-1, for more information.
3. If necessary, install BladeManager.

   BladeManager is included in the IBM Informix installation bundle. If you did not install BladeManager when you installed Informix, run `BladeMgr\Setup.exe` from the installation media.

To start BladeManager, select **Start** > **Programs** > ***Informix program group name*** > **BladeManager** or double-click the **BladeManager** icon in the Informix program group. To see a particular page in the application window, click its tab.

© Copyright IBM Corp. 1996, 2013

Managing DataBlade modules

To manage DataBlade modules, use the Databases page, as shown in the following figure.

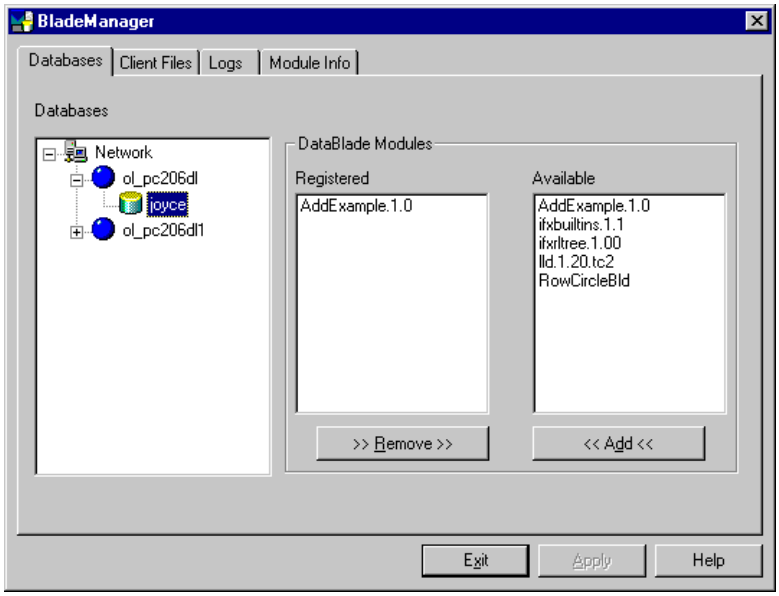

Figure 3-1. Databases page

Managing DataBlade modules consists of tasks described in the following topics.

Connecting to a database

After you install a DataBlade module, you register it in each database in which you want to use it. To register a DataBlade module, you must first connect to that database.

See the *IBM Informix Administrator's Guide* for information about the connect and resource permissions you need to connect to the database.

Tip: You can be sure you have the right permissions if you run BladeManager as the default user for the database server. You can tell you are the default user if you are not prompted for a user name and password when you attempt to connect to the database. You can use the **Setnet32** utility to specify the user name and password you want to be the default for a particular database server. Make sure you restart BladeManager whenever you change settings in **Setnet32**.

To connect to a database:

1. To see a list of available database servers, in the **Databases** list box on the **Databases** page, click the expander button next to the network and database server icons.
2. Click the name of the database to which you want to connect.
3. If the User Login dialog box appears, type a user name and password that have the required permissions for the database.
4. Click **OK**.

After you connect, BladeManager displays the registered and available DataBlade modules for that database.

The first time BladeManager connects to a database, BladeManager prepares the installed DataBlade modules for registration and generates a log file. During the preparation, BladeManager gathers the DataBlade module information that appears on the BladeManager pages. If the preparation of a DataBlade module fails, the DataBlade module does not appear in the Available list box. Check the log file for information about preparation failures (see "Viewing log files" on page 3-6) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Registering a DataBlade module

When BladeManager registers a DataBlade module, it executes a set of SQL statements to register each database object in the module. Registration is equivalent to creating database objects individually with the SQL CREATE statement.

You must have resource permissions on the database to register a DataBlade module in it. Additionally, if the server is configured so that the EXTEND role is needed to add UDRs and UDTs, then you must be granted the EXTEND role by a DBSA (typically, user **informix**).

To register a DataBlade module:

1. On the Databases page, select the database in which you want to register a module.
2. In the Available list box, select the module you want to register.
3. Click **Add**.
4. Click **Apply**.

If registration fails, BladeManager returns the database to its prior state. To see the SQL statements that failed, look at the corresponding log file (see "Viewing log files" on page 3-6) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Some DataBlade modules depend on one or more *interfaces*. An interface is like a contract between DataBlade modules: the DataBlade module that requires the interface depends on the functionality of the DataBlade module that provides the interface.

When you register a DataBlade module with an interface dependency, BladeManager verifies that one of the DataBlade modules that provides that interface is registered in the database. If it is, registration continues. If it is not, BladeManager displays the Modules with Missing Interface dialog box; select one of the DataBlade modules and click **OK**.

Important: BladeManager does not verify the integrity of the DataBlade modules that provide a required interface; BladeManager does not check for the presence of the required database objects.

Upgrading a DataBlade module

To upgrade a DataBlade module, use BladeManager to register a new version of the module. When you register the new version, BladeManager will automatically unregister the old version.

Restriction: You cannot use the following procedure to upgrade or downgrade some versions of DataBlade modules. For instructions on which versions can be upgraded or downgraded, see the release notes for the DataBlade module.

To upgrade a DataBlade module:

1. On the Databases page, select the database in which you want to upgrade a DataBlade module (see Figure 3-1 on page 3-2).
2. In the Available list box, select the module you want to upgrade.
3. Click **Add**.

   The Registered list box shows the version of the module with the new version in parentheses to indicate that the current version will be upgraded.
4. Click **Apply**.

   After a successful upgrade, the Registered list box shows only the new version, along with any other DataBlade modules registered in the database.

If the upgrade fails, BladeManager returns the database to its prior state. To see the SQL statements that failed, look at the corresponding log file (see "Viewing log files" on page 3-6) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

To downgrade a DataBlade module, use this same procedure to replace the DataBlade module with a lower version, which appears in the Available list box.

Unregistering a DataBlade module

When BladeManager unregisters a DataBlade module, it removes each object of the module from the database by using SQL DROP statements.

Restriction: BladeManager does not unregister a DataBlade module that provides a required interface for other DataBlade modules or database objects.

To unregister a DataBlade module:

1. On the Databases page, select the database from which you want to unregister a module (see Figure 3-1 on page 3-2).
2. In the Registered list box, select the module you want to unregister.
3. Click **Remove**.
4. Click **Apply**.

If the unregistration fails, BladeManager returns the database to its prior state. To see the SQL statements that failed, look at the corresponding log file (see "Viewing log files" on page 3-6) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Managing client files

Some DataBlade modules are shipped with files that are required on client computers. These client files can include graphical user interfaces to view data or tools to query or search the database. When you install the DataBlade module, the client files are placed on the database server. You can install and uninstall these client files on the client computer that runs BladeManager.

You can manage client files with the Client Files page, as shown in the following figure.

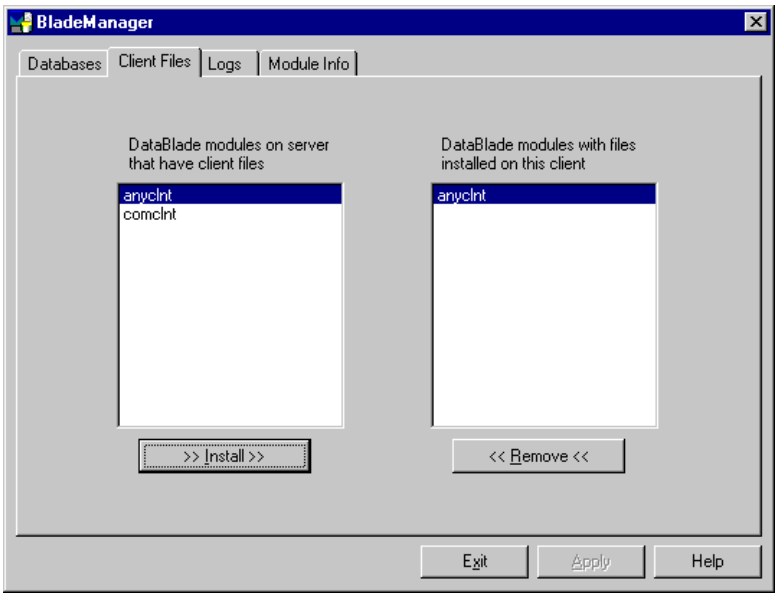

Figure 3-2. Client files page

The left list box shows the names of all the DataBlade modules on the current server that include client files that are appropriate for the current client computer. DataBlade modules can contain additional client files for other operating systems. The right list box shows the names of all DataBlade modules that have client files installed on the current client computer.

Installing client files

You must install client files individually on each computer on which you require them.

Typically, client files are installed on the client computer in the `$INFORMIXDIR/extend/`*`datablade`*`/client` directory for your IBM Informix client products, where *datablade* is the name of the DataBlade module. However, some DataBlade modules might install files in other directories. To install client files, you must have permission to write to the directory in which the client files are installed.

To install the client files:

1. On the Client Files page, select the DataBlade module that contains the client files you want to install from the left list box.
2. Click **Install**.

A client file installation might require processing after BladeManager has copied the files to your computer. For example, you might have to run an install script or `setup.exe` program before you can use the client files. For instructions, see the release notes for the DataBlade module.

Uninstalling client files

You must uninstall client files from each computer from which you want them removed.

To uninstall client files, you must have permission to write to the directory in which the client files reside.

To remove client files:

1. On the Client Files page, select a DataBlade module in the right list box.
2. Click **Remove**.

A client file uninstallation might require processing before or after BladeManager has removed the files from your computer. For example, you might have to run an uninstall script or program before the client files are completely removed. For instructions, see the release notes for the DataBlade module.

Viewing log files

BladeManager generates a log file when you prepare a database for registration and when you register, upgrade, or unregister a DataBlade module. If one of these tasks fails, the log file lists the specific SQL statement that failed. Log files also list whether the failure was expected or unexpected, and they show the error generated by the SQL command.

An example of an expected error is an error issued when a DataBlade module contains an SQL statement to create a table, but that table already exists. When BladeManager receives an unexpected error, it halts the operation and returns the database to its prior state.

Log files are numbered consecutively and contain a time stamp. You should periodically delete log files to free disk space.

View and delete log files on the Logs page, as shown in the following figure.

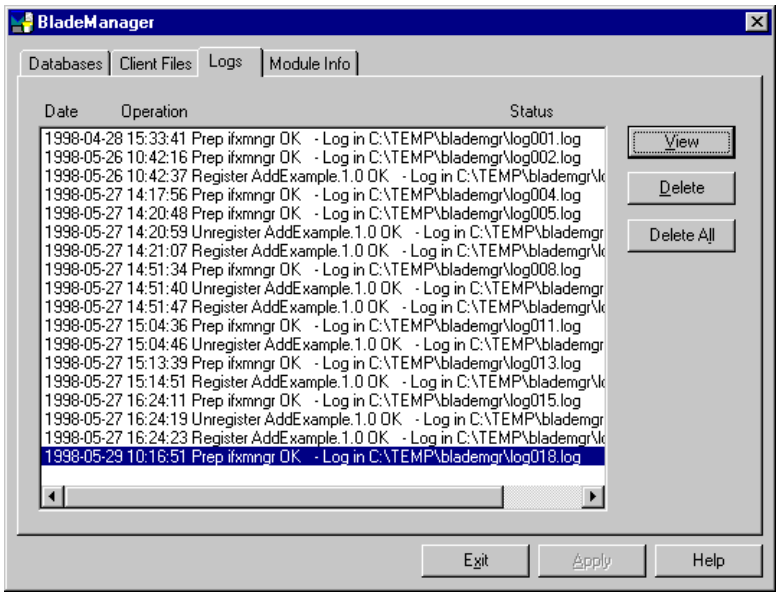

Figure 3-3. Logs page

Viewing a log file

To view a log file:

1. On the Logs page, select the log file you want to view.
2. Click **View**.

Deleting a log file

To delete a log file:

1. On the Logs page, select the log file you want to delete.
2. Click **Delete**.

View module information

Each DataBlade module has content and vendor information. After BladeManager prepares the DataBlade modules for a database, you can view the information for all DataBlade modules.

The Module Info page displays information about DataBlade modules installed in the database selected on the Databases page.

To view module information, on the Module Info page, select a DataBlade module in the DataBlade modules list box, as shown in the following figure.

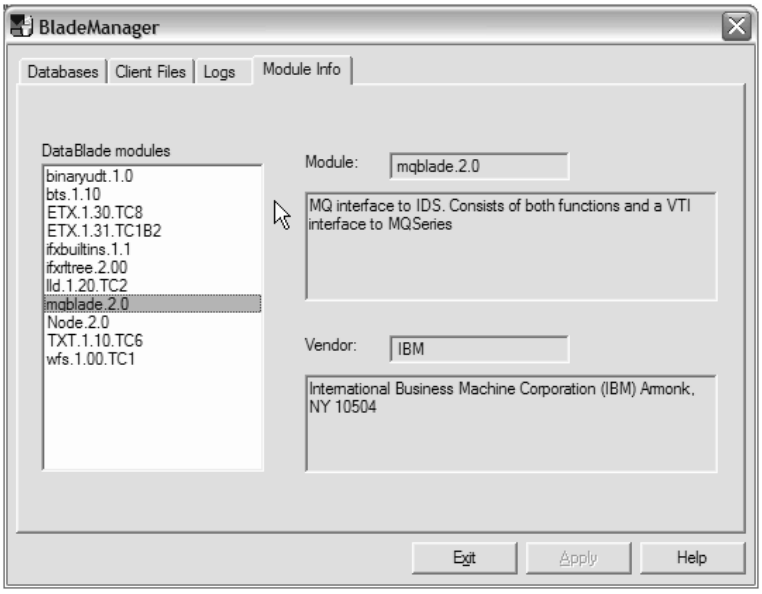

Figure 3-4. Module Info page

The Module box shows the full name and version of the selected module, and it might display a description. The Vendor box shows information about the vendor of the DataBlade module.

Chapter 4. Registering with the BladeManager command-line interface

To use a DataBlade module in a database, you must first register the DataBlade module in the database. *Registration* is the process of executing the SQL statements that create the DataBlade module database objects and identify the DataBlade module shared object file or dynamic link library to the database server. You use BladeManager to register DataBlade modules.

Before you can use BladeManager, you must complete the tasks listed in "Preparing to use the BladeManager command-line interface."

These topics describe how to use the BladeManager command-line interface for DataBlade module registration on UNIX and Windows. You can also call the **SYSBldPrepare( )** function to register DataBlade modules on UNIX and Windows, or use the graphic interface of BladeManager to register DataBlade modules on Windows. For details of DataBlade module registration by using SQL function calls, see Chapter 2, "Registering with the SYSBldPrepare( ) function," on page 2-1. For details of DataBlade module registration on Windows using the graphic interface of BladeManager, see Chapter 3, "Registering with the BladeManager graphical user interface," on page 3-1.

The BladeManager command-line interface provides commands to perform tasks.

Preparing to use the BladeManager command-line interface

This topic lists the prerequisite tasks for using BladeManager to register DataBlade modules.

To prepare to use the BladeManager command-line interface:

1. Configure your IBM Informix environment in one of the following ways:
   - On UNIX, set **INFORMIXDIR**, **PATH**, **LD_LIBRARY_PATH**, **ONCONFIG**, and **INFORMIXSERVER**. For information about setting these environment variables, see the *IBM Informix Administrator's Guide*.
   - On Windows, set **INFORMIXDIR** and **INFORMIXSERVER**. For information about setting these environment variables, see the *IBM Informix Installation Guide for Windows*.
2. 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 *IBM Informix Administrator's Reference*.
3. Install DataBlade modules. See Chapter 1, "Installing DataBlade modules," on page 1-1, for more information.
4. Install BladeManager, if necessary.

   BladeManager is included in the installation for your database server on UNIX.

   BladeManager is included in the installation bundle for your database server on Windows. Alternatively, you can install BladeManager on Windows by running the `BladeMgr\Setup.exe` program in your database server product media.

© Copyright IBM Corp. 1996, 2013

Start and stop BladeManager

To start BladeManager, enter the **blademgr** command at the UNIX or the MS-DOS command-line prompt.

Your screen displays a prompt consisting of the value of the current database server (the value of the **INFORMIXSERVER** environment variable when you start BladeManager) and an angle bracket (>). To execute BladeManager commands, enter the command name.

Important: If BladeManager fails to execute, make sure that IBM Informix environment variables are set to run Informix database utilities. See "Preparing to use the BladeManager command-line interface" on page 4-1 for more information.

To exit BladeManager, enter one of the following commands:

- **bye**
- **exit**
- **quit**
- an end-of-file key sequence, such as **CTRL-D** or **CTRL-Z**

Obtain help for commands

To see a list of BladeManager commands and their syntax, enter one of the following commands:

- **help**
- **?**

Set confirmation

If confirmation is on, BladeManager prompts you to confirm the following tasks:

- Register a DataBlade module with the **register** command
- Unregister a DataBlade module with the **unregister** command
- Install a client file with the **add client** command
- Uninstall a client file with the **del client** command

When confirmation is off, BladeManager issues no confirmation prompt when you execute these commands. You might want to turn confirmation off when you use BladeManager in batch operations, so you can send commands to BladeManager from a file.

To turn confirmation on, enter the following command:

```
set confirm on
```

To turn confirmation off, enter the following command:

```
set confirm off
```

Execute BladeManager commands automatically at startup

When BladeManager starts, it searches for the batch file `blademgr.run` in the current directory. If BladeManager finds the file, it executes any commands in it before it displays the prompt that allows you to enter commands. You can edit this file and place commands in it that are automatically executed whenever BladeManager starts.

For example, if you do not want to be prompted for confirmation each time you register a DataBlade module, you can put the **set confirm off** command in `blademgr.run` to turn confirmation off automatically whenever you run BladeManager.

If you include multiple commands in `blademgr.run`, separate each command with a carriage return.

Execute multiple BladeManager commands

To execute a series of BladeManager commands using shell redirection, enter the following command:

```
blademgr < filename.txt
```

The *filename.txt* specification in the example represents the name of a text file that contains one or more BladeManager commands, each on a separate line.

Preparation for database-server reversion

To revert to an earlier version of your database server, you must unregister all DataBlade modules and then remove all database objects that were added since the database and database server were upgraded. BladeManager adds its objects to a database whenever you connect to a database with BladeManager.

Important: You cannot revert to an earlier database-server version if database objects created before the database server was upgraded are have been altered.

You must use BladeManager to remove SQL objects that were created during registration of the ifxmngr DataBlade module. After you remove the SQL objects, run the following command:

```
unprep database_name
```

For more information about server reversion, see the *IBM Informix Migration Guide*.

Connection information

After you install a DataBlade module, you register it in each database in which you want to use it. To register a DataBlade module in a database, you must first connect to that database.

You need connect and resource permissions to connect to the database and register DataBlade modules. See the *IBM Informix Administrator's Guide* for more information about permissions.

Before BladeManager can connect to a database, your IBM Informix environment must be properly configured on the computer running the database server and on the client computer.

Connect to your database server

When you start BladeManager, it uses default values for your user name, password, and database server unless you explicitly issue commands to change them.

On UNIX, the default user name and password are defined as your operating system login and password. The default database server is defined as the value of

the **INFORMIXSERVER** environment variable. For information about setting these values, see *IBM Informix Administrator's Guide*.

On Windows, the default values for user name, password, and database server are listed in the **Setnet32** utility. The default database server is defined as the value of the **INFORMIXSERVER** environment variable. For information about setting these values, see *IBM Informix Client Products Installation Guide*.

You can change your connection to an IBM Informix database server as follows:

- To see a list of available database servers, enter the following command:

  ```
  show servers
  ```

- To connect to a specified database server, enter the following command:

  ```
  set server server_name
  ```

- To connect as a different user, enter the following command:

  ```
  set user user_name
  ```

At the password prompt, enter the password. The user name and password are not validated until you attempt to connect to a database.

Tip: To avoid possible permissions problems, run BladeManager as the default user for the database server.

Connect to databases

BladeManager commands to register, unregister, list registered, and show available DataBlade modules operate on a specific database.

After you start BladeManager, you can connect to a database, as follows:

- To display a list of databases available to a database server, connect to the database server and enter the following command:

  ```
  show databases
  ```

- To connect to a database, execute one of the following commands:
  - `list database_name`
  - `register module_name database_name`
  - `unregister module_name database_name`

  In the preceding example commands, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number. See "Register a DataBlade module" for information about the **list** and **register** commands. See "Unregister a DataBlade module" on page 4-6 for information about the **unregister** command.

The first time BladeManager connects to a database, it prepares the installed DataBlade modules for registration and generates a log file. If BladeManager fails to connect to a database or preparation fails, look at the appropriate log file (see "View log files" on page 4-8) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Register a DataBlade module

When BladeManager registers a DataBlade module, it executes a set of SQL statements to register each database object in the module. Registration is equivalent to creating database objects individually with the SQL CREATE statement.

You must have resource permissions on the database to register a DataBlade module in it. Additionally, if the server is configured so that the EXTEND role is needed to add UDRs and UDTs, then you must be granted the EXTEND role by a DBSA (typically, user **informix**).

After you connect to a database, you can register a DataBlade module, as follows:

- To display a list of DataBlade modules installed on the database server that are available for registration, enter the following command:

  ```
  show modules
  ```

  Modules that contain client files display the letter `c` after the module name.
- To display a list of DataBlade modules registered in the specified database, enter the following command:

  ```
  list database_name
  ```
- To register a DataBlade module in the specified database, enter the following command:

  ```
  register module_name database_name
  ```

  In the preceding example command, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number.

If BladeManager is not currently connected to the database specified in the **register** command, it connects to the new database, prepares all available DataBlade modules, and displays a message about the success or failure of the preparation before continuing with registration.

If the registration of a module fails, BladeManager returns the database to its prior state. To see the SQL statements that failed, look at the corresponding log file (see "View log files" on page 4-8) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Some DataBlade modules depend on one or more *interfaces*. An interface is like a contract between DataBlade modules: the DataBlade module that requires the interface depends on the functionality of the DataBlade module that provides the interface.

When you register a DataBlade module, BladeManager verifies that one of the DataBlade modules that provides the interface required by your module is already registered in the database. If it is, registration continues. If it is not, BladeManager prompts you to register the DataBlade module providing the interface.

Important: BladeManager does not verify the integrity of the DataBlade modules that provide a required interface. BladeManager does not check for the presence of the required database objects.

Upgrade a DataBlade module

To upgrade a DataBlade module, use BladeManager to register a new version of the module. When you register the new version, BladeManager automatically unregisters the old version.

Restriction: You cannot use the following procedure to upgrade or downgrade some versions of DataBlade modules. For instructions on which versions can be upgraded or downgraded, see the release notes for the DataBlade module.

To upgrade or downgrade a DataBlade module in the specified database, enter the following command:

```
register module_name database_name
```

In the preceding example command, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number. BladeManager warns you that you are upgrading or downgrading a DataBlade module.

If BladeManager is not currently connected to the database specified in the **register** command, it connects to the new database, prepares all available DataBlade modules, and displays a message about the success or failure of the preparation before continuing with registration.

If the upgrade of a module fails, BladeManager returns the database to its prior state. To see the SQL statements that failed, look at the corresponding log file (see "View log files" on page 4-8) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Unregister a DataBlade module

When BladeManager unregisters a DataBlade module, it removes each element of the DataBlade module from the database using SQL DROP statements.

Restriction: BladeManager does not unregister a DataBlade module that provides a required interface for other DataBlade modules or database objects.

To unregister a DataBlade module in the specified database, enter the following command:

```
unregister module_name database_name
```

In the preceding example, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number. If BladeManager is not currently connected to the database specified in the **unregister** command, it connects to the new database, prepares all available DataBlade modules, and displays a message about the success or failure of the preparation before continuing with the **unregister** command.

If the unregistration of a module fails, BladeManager returns the database to its prior state. To see the SQL statements that failed, look at the corresponding log file (see "View log files" on page 4-8) and see Appendix A, "Troubleshooting registration problems," on page A-1, for possible solutions.

Manage client files

Some DataBlade modules are shipped with files that are required on client computers. These client files can include command-line interfaces to view data or tools to query or search the database. When you install the DataBlade module, the client files are placed on the database server along with the elements of the module. You can install and uninstall these client files on the client computer running BladeManager.

To display a list of DataBlade modules installed on the database server to which BladeManager is connected that have client files installed on the computer on which BladeManager is running, enter the following command:

```
show client
```

Install client files

You must install client files individually on every computer on which you want them installed by running BladeManager on each computer.

Typically, client files are installed on the client computer in the `$INFORMIXDIR/extend/`*`datablade`*`/client` directory for your IBM Informix client products, where *datablade* is the name of the DataBlade module. However, some DataBlade modules might install files in other directories. To install client files, you must have permission to write to the directory in which the client files are installed.

To install client files for a specific DataBlade module, enter the following command:

```
add client module_name
```

In the preceding example, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number.

A client file installation might require processing after BladeManager has copied the files to your computer. For example, you might have to run an install script or `setup.exe` program before you can use the client files. For instructions, see the release notes for that DataBlade module.

Uninstall client files

You must uninstall client files individually from each computer from which you want them removed.

You must have permission to write to the directory in which the client files reside.

To uninstall client files for a specific DataBlade module, enter the following command:

```
del client module_name
```

In the preceding example, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number.

A client file uninstallation might require processing before or after BladeManager has removed the files from your computer. For example, you might have to run an uninstall script or program before the client files are completely removed. For instructions, see the release notes for that DataBlade module.

View log files

BladeManager generates a log file whenever you prepare, register, upgrade, or unregister a DataBlade module. If one of these tasks fails, the log file can point to the particular SQL statement that failed. Log files also list whether the failure was expected or unexpected and show the text of the error generated by the SQL command.

An example of an expected error is an error issued when a DataBlade module contains an SQL statement to create a table, but that table already exists. When BladeManager receives an unexpected error, it halts the operation and returns the database to its prior state.

BladeManager stores log files in one of the following directories:

- For UNIX, `/tmp/blademgr/uid`, where *uid* is your UNIX user ID
- For Windows, `%TEMP%\blademgr`

Periodically delete files from the log directory to free disk space.

You can manage log files by performing the following tasks:

- To view the list of log files, enter the following command:

  ```
  show log
  ```

  BladeManager lists all available log files. You can see only log files created while BladeManager was running with your user ID.
- To see a particular log file, display the list of log files and enter the log file number.
- To see the most recent log file for the current session, enter the following command:

  ```
  show last log
  ```
- To delete log files for your user ID, enter the following command:

  ```
  del logs
  ```

  BladeManager prompts you to continue with the removal of the log files from the operating system.

View module information

Each DataBlade module has content and vendor information.

To display information about a particular DataBlade module, enter the following command:

```
info module_name
```

In the preceding example, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number.

If you are not connected to a database, use the following command to establish a connection to the specified database before executing the **info** command:

```
list database_name
```

Command reference

The following table describes the BladeManager commands in detail. The commands are listed in alphabetic order.

In the examples of each command, *module_name* represents the name of the DataBlade module directory. These names typically follow the form of the DataBlade module name followed by the version number. See "Uninstall client files" on page 4-7 for more information.

Command	Description
add client	The **add client** command installs the specified client files for the DataBlade module on the client computer running BladeManager: `add client module_name` See "Install client files" on page 4-7 for more information.
bye	The **bye** command closes BladeManager and returns you to the operating system prompt.
del client	The **del client** command removes the specified client files for the DataBlade module from the client computer running BladeManager: `del client module_name` See "Uninstall client files" on page 4-7 for more information.
del logs	The **del logs** command deletes the `/tmp/blademgr/uid` directory on UNIX or the `%TEMP%\blademgr` directory on Windows. The directory contains the log files from your BladeManager session. See "View log files" on page 4-8 for more information.
exit	The **exit** command closes BladeManager and returns you to the operating system prompt.
help	The **help** command displays a list of all the BladeManager commands and syntax.
info	The **info** command displays vendor-supplied information about the specified DataBlade module: `info module_name` See "View module information" on page 4-8 for more information.
list	The **list** command displays all the DataBlade modules already registered with the specified database: `list database_name` See "View module information" on page 4-8 for more information.

Command	Description
quit	The **quit** command closes BladeManager and returns you to the operating system prompt.
register	The **register** command registers the specified DataBlade module in the specified database: `register module_name database_name` See "Register a DataBlade module" on page 4-4 for more information.
set confirm	The **set confirm** command toggles confirmation for the **register**, **unregister**, **add client**, or **del client** commands on and off: `set confirm on | off` See "Set confirmation" on page 4-2 for more information.
set server	The **set server** command connects BladeManager to the specified database server: `set server server_name` See "Connect to your database server" on page 4-3 for more information.
set user	The **set user** command sets the user for the current session: `set user user_name` BladeManager prompts you for a password. See "Connect to your database server" on page 4-3 for more information.
show client	The **show client** command displays a list of the DataBlade modules on the current database server that have files installed on the current client computer.
show databases	The **show databases** command displays a list of the databases on the current database server.
show last log	The **show last log** command displays the most recent log file for the current session. The **show last log** command does not return a log file if you have not executed a command during the current BladeManager session.
show log	The **show log** command lists the available log files and allows you to view the contents of a specific log file. After BladeManager lists all the log files that are available for viewing, it prompts you for the number of the particular log file whose contents you want to view.

Command	Description
show modules	The **show modules** command displays a list of the DataBlade modules available on the database server. Modules that contain client files display the letter `c` after the module name.
show servers	The **show servers** command displays a list of the available servers.
unprep	The **unprep** command removes BladeManager from the specified database to allow you to revert from one version of your database server to an earlier version: `unprep` *`database_name`* See "Preparation for database-server reversion" on page 4-3 for more information.
unregister	The **unregister** command unregisters the specified DataBlade module from the specified database: `unregister` *`module_name database_name`* See "Unregister a DataBlade module" on page 4-6 for more information.
?	The **?** (question mark) command displays a list of all the BladeManager commands and syntax.

Appendix A. 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.

IBM 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 ('geodetic.3.12', 'create');

(U0001) - registerBlade - Unable to register geodetic.3.12
– 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 IBM 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.

© Copyright IBM Corp. 1996, 2013

IBM 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 IBM 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 IBM 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.

Appendix B. Accessibility

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

Accessibility features for IBM Informix products

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

Accessibility features

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

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

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

Related accessibility information

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

IBM and accessibility

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

Dotted decimal syntax diagrams

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

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

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

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

© Copyright IBM Corp. 1996, 2013

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

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

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

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

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

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

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

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

Notes:

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

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

Notices

This information was developed for products and services offered in the U.S.A.

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

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

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

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

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

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

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

Any references in this information to non-IBM websites are provided for convenience only and do not in any manner serve as an endorsement of those websites. The materials at those websites are not part of the materials for this IBM product and use of those websites is at your own risk.

© Copyright IBM Corp. 1996, 2013

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

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

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy,

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

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

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

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

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

Trademarks

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

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

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

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

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

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

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

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

Index

Special characters

A

B

C

© Copyright IBM Corp. 1996, 2013

D

E

G

H

I

L

M

P

Q

R

S

T

U

V

W

Printed in USA

GC27-4517-00