Informix Product Family
Informix
Version 11.70

IBM Informix Embeddability Guide

SC27-3528-04

Informix Product Family
Informix
Version 11.70

IBM Informix Embeddability Guide

SC27-3528-04

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

Edition

This edition replaces SC27-3528-03.

Contents

Introduction

The information in this guide is intended for application programmers.

About this publication

This publication describes how to embed IBM® Informix® in your application.

What's new for embeddability in IBM Informix, Version 11.70

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

For a complete list of what's new in this release, see the release notes or the information center at http://publib.boulder.ibm.com/infocenter/idshelp/v117/topic/com.ibm.po.doc/new_features.htm.

Table 1. What's new for embeddability in Informix 11.70.xC3

Overview	Reference
Managing message logs in embedded and enterprise environments You can use Scheduler tasks to reduce the size of message log files by automatically truncating or deleting the log files or by configuring automatic file rotation. Additionally, you can use the related **ph_threshold table** parameters to specify the maximum number of message log files to retain. These tasks and parameters are useful for embedded applications because they reduce DBA or system administrator requirements for managing the log files. You can also use SQL administration API commands to manage the size of the log files on demand, as necessary.	"Manage message logs in an embedded environment" on page 4-1
Configuring the server response to low memory You can configure the actions that the server takes to continue processing when memory is critically low. You can specify the criteria for terminating sessions based on idle time, memory usage, and other factors so that the targeted application can continue and avoid out-of-memory problems. Configuring the low memory response is useful for embedded applications that have memory limitations.	"Maintain a targeted amount of memory in embedded applications" on page 4-5
Reserving memory for critical activities You can enable the server to reserve a specific amount of memory for use when critical activities (such as rollback activities) are needed and the server has limited free memory. If you enable the new LOW_MEMORY_RESERVE configuration parameter by setting it to a specified value in kilobytes, the critical activities can complete even when you get out-of-memory errors. You can also dynamically adjust the value of the LOW_MEMORY_RESERVE configuration parameter with the **onmode -wm** or **-wf** command.	"Reserve memory for critical activities in embedded environments" on page 4-5

Table 2. What's new for embeddability in Informix 11.70.xC2

Overview	Reference
Embedding Informix software without root privileges (UNIX, Linux) You can install Informix software without having root privileges. When you install the product without root privileges, the user account that performs the installation becomes the database server administrator (DBSA) for that installation. You can then copy the non-root installation to and deploy it on other computers as part of a deeply embedded database server scenario. A non-root installation does not support some major features such as Enterprise Replication, distributed connections, and high-availability clusters.	"Customize a silent installation" on page 1-1
Deploying an RPM image of Informix software (Linux) When you install Informix you can customize an RPM Package Manager image to exclude database server and client products that you do not plan to use. You can then embed the configured installation to multiple Linux computers that support RPM. By selectively removing features that you do not need, you can reduce the size of the distributable image. Also, you can deploy the image without the system resource demands of the installation application.	"Deploying Informix and client products with RPM-based installation (Linux)" on page 2-22

Table 3. What's new for embeddability in Informix 11.70.xC1

Overview	Reference
Information about embedding Informix instances In the previous release, information about deploying embedded instances of Informix was documented along with the information about installing the database server. In this release, information about deploying embedded Informix instances can be found in the Embedding Informix section of the online information center or in the new *IBM Informix Embeddability Guide*.	See Chapter 1, "Overview of embedding Informix products," on page 1-1 for an outline of how the information in the embeddability documentation is structured.
Enhanced utility for deploying Informix instances In this release it is easier to use the deployment utility (**ifxdeploy**) to rapidly deploy a configured database server instance to multiple computers. The **-start** option deploys and starts an instance in a single operation so that you can silently deploy a database server. The **-autorecommend** option calculates optimal values for database server configuration parameters based on your planned usage for the database server and the host environment. The `ifxdeploy.conf` file contains new parameters so that you can run the deployment utility with fewer command-line options.	"The ifxdeploy command: The deployment utility" on page 2-8

Table 3. What's new for embeddability in Informix 11.70.xC1 (continued)

Overview	Reference
Deployment assistant simplifies snapshot capture and configuration In past releases you had to manually create a snapshot. In this release you can use the built-in intelligence of the deployment assistant to capture and configure an Informix snapshot more easily. Run the **ifxdeployassist** command to start the deployment assistant interface, which prompts you for the required information to capture the instance. Use the **-c** option if you want to pass command options in a scripting environment instead of being prompted by the deployment assistant interface. You must use the interface instead of the command line if you want to capture a reduced-footprint snapshot that contains only specific features.	"Creating a snapshot with the deployment assistant" on page 2-2
Enhanced ability to compress and to extract compressed snapshots The deployment assistant supports the following archive formats: BZIP2, GZIP, TAR, and ZIP. The deployment utility automatically extracts snapshots that were compressed in BZIP2, GZIP, TAR, and ZIP formats. In the previous release you had to specify the **-extractcmd** option to extract BZIP2 and GZIP formats.	"Creating a snapshot with the deployment assistant" on page 2-2 "The ifxdeploy command: The deployment utility" on page 2-8
Tutorial to deploy and embed Informix Follow the step-by-step tutorial to deploy a preconfigured Informix instance with a minimal footprint on multiple computers. The steps describe how to use the deployment assistant to configure and create a snapshot of the instance and how to use the deployment utility to deploy the instance in an embedded environment. A sample script (`ifx_silent_deploy`) is provided for automating the process.	Chapter 3, "Informix embeddability deployment tutorial," on page 3-1
New editions and product names IBM Informix Dynamic Server editions were withdrawn and new Informix editions are available. Some products were also renamed. The publications in the Informix library pertain to the following products: • IBM Informix database server, formerly known as IBM Informix Dynamic Server (IDS) • IBM OpenAdmin Tool (OAT) for Informix, formerly known as OpenAdmin Tool for Informix Dynamic Server (IDS) • IBM Informix SQL Warehousing Tool, formerly known as Informix Warehouse Feature	For more information about the Informix product family, go to http://www.ibm.com/software/data/informix/.

Types of users

This publication is for application developer who need to embed IBM Informix in an application for redistribution.

Example code conventions

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

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

```
CONNECT TO stores_demo
...

DELETE FROM customer
   WHERE customer_num = 121
...

COMMIT WORK
DISCONNECT CURRENT
```

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

Tip: Ellipsis points in a code example indicate that more code would be added in a full application, but it is not necessary to show it to describe the concept 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 or install the product documentation from the Quick Start CD that is shipped with Informix products. To get the most current information, see the Informix information centers at ibm.com®. You can access the information centers and other Informix technical information such as 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.

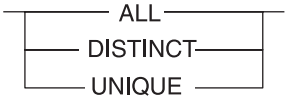
The IBM Informix Geodetic DataBlade® Module supports a subset of the data types from the *Spatial Data Transfer Standard (SDTS)—Federal Information Processing Standard 173*, as referenced by the document *Content Standard for Geospatial Metadata*, Federal Geographic Data Committee, June 8, 1994 (FGDC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

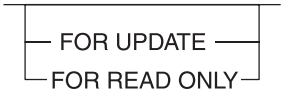
Syntax diagrams

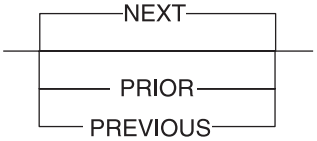
Syntax diagrams use special components to describe the syntax for statements and commands.

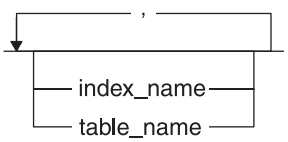
Table 4. Syntax Diagram Components

Component represented in PDF	Component represented in HTML	Meaning
	`>>----------`	Statement begins.
	`---------->`	Statement continues on next line.
	`>----------`	Statement continues from previous line.
	`----------><`	Statement ends.
	`--------SELECT----------`	Required item.
	`--+-----------------+---` `  '------LOCAL------'`	Optional item.

	`---+-----ALL-------+---`
	`   +--DISTINCT-----+`
	`   '---UNIQUE------'`
		Required item with choice. Only one item must be present.

	`---+------------------+---`
	`    +--FOR UPDATE-----+`
	`    '--FOR READ ONLY--'`
		Optional items with choice are shown below the main line, one of which you might specify.

	`    .---NEXT---------.`
	`----+----------------+---`
	`    +---PRIOR--------+`
	`    '---PREVIOUS-----'`
		The values below the main line are optional, one of which you might specify. If you do not specify an item, the value above the line is used by default.

	`  .-------,-----------.`
	`  V                   |`
	`---+-----------------+---`
	`    +---index_name---+`
	`    '---table_name---'`
		Optional items. Several items are allowed; a comma must precede each repetition.

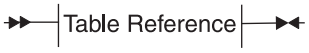
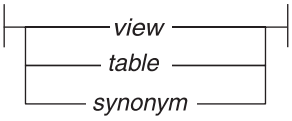
	`>>-| Table Reference |-><`	Reference to a syntax segment.

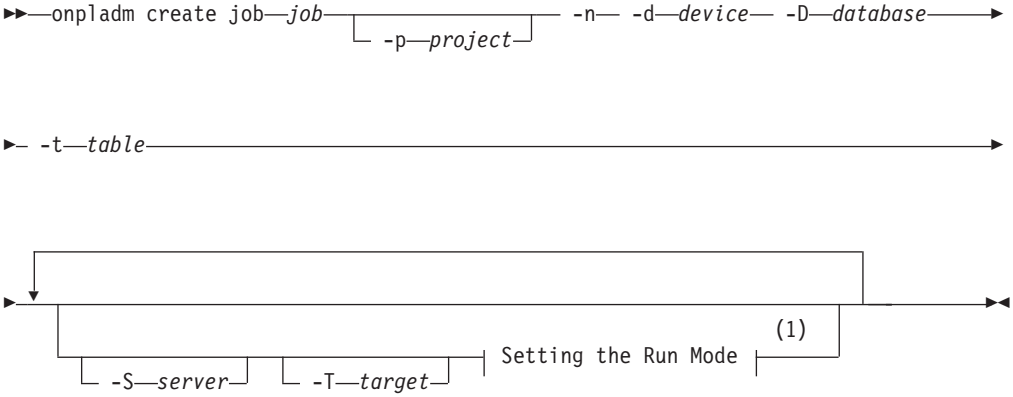
Table Reference
 
	`Table Reference`
	`|--+-----view--------+--|`
	`   +------table------+`
	`   '----synonym------'`
		Syntax segment.

How to read a command-line syntax diagram

Command-line syntax diagrams use similar elements to those of other syntax diagrams.

Some of the elements are listed in the table in Syntax Diagrams.

Creating a no-conversion job

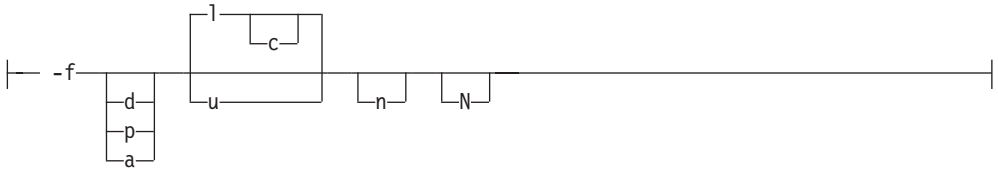


Notes:

1 See page Z-1

This diagram has a segment named "Setting the Run Mode," which according to the diagram footnote is on page Z-1. If this was an actual cross-reference, you would find this segment on the first page of Appendix Z. Instead, this segment is shown in the following segment diagram. Notice that the diagram uses segment start and end components.

Setting the run mode:



To see how to construct a command correctly, start at the upper left of the main diagram. Follow the diagram to the right, including the elements that you want. The elements in this diagram are case-sensitive because they illustrate utility syntax. Other types of syntax, such as SQL, are not case-sensitive.

The Creating a No-Conversion Job diagram illustrates the following steps:

1. Type **onpladm create job** and then the name of the job.
2. Optionally, type **-p** and then the name of the project.
3. Type the following required elements:
   - **-n**
   - **-d** and the name of the device
   - **-D** and the name of the database
   - **-t** and the name of the table

4. Optionally, you can choose one or more of the following elements and repeat them an arbitrary number of times:
   - **-S** and the server name
   - **-T** and the target server name
   - The run mode. To set the run mode, follow the Setting the Run Mode segment diagram to type **-f**, optionally type **d**, **p**, or **a**, and then optionally type **l** or **u**.
5. Follow the diagram to the terminator.

Keywords and punctuation

Keywords are words reserved for statements and all commands except system-level commands.

When a keyword appears in a syntax diagram, it is shown in uppercase letters. When you use a keyword in a command, you can write it in uppercase or lowercase letters, but you must spell the keyword exactly as it appears in the syntax diagram.

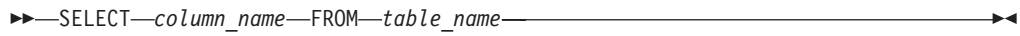
You must also use any punctuation in your statements and commands exactly as shown in the syntax diagrams.

Identifiers and names

Variables serve as placeholders for identifiers and names in the syntax diagrams and examples.

You can replace a variable with an arbitrary name, identifier, or literal, depending on the context. Variables are also used to represent complex syntax elements that are expanded in additional syntax diagrams. When a variable appears in a syntax diagram, an example, or text, it is shown in *lowercase italic*.

The following syntax diagram uses variables to illustrate the general form of a simple SELECT statement.



When you write a SELECT statement of this form, you replace the variables *column_name* and *table_name* with the name of a specific column and table.

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. Overview of embedding Informix products

You can embed Informix products in your applications by taking advantage of customizable deployment options and configurable administration features. You can use the embeddability features in combination with other product features for installation footprint reduction and high-availability clustering to achieve a quicker and easier end-to-end deployment solution.

Customize the size of Informix

If your application runs in a low-memory environment, you might need to minimize the size of IBM Informix.

You can customize the size of IBM Informix during installation and during snapshot creation.

You can use the Deployment wizard during installation to exclude parts of the Informix product that your application does not need. The installation program ensures that all dependent components are automatically selected. The installation program shows you what the size of the product will be after installation. The minimum size of an Informix installation is approximately 100 MB.

You can use the Deployment Assistant to exclude parts of the product from the snapshot of an existing Informix instance that you intend to deploy embedded in your application. The Deployment Assistant does not enforce dependencies. You can also choose which Informix client products and which dbspaces being used by the Informix instance to include in the snapshot.

Customize a silent installation

You can create a silent installation script, customized for your requirements, that you can use multiple times.

To create a custom silent installation script, you install IBM Informix using the GUI- or console-mode to capture installation settings that you plan to reuse in multiple locations (often different host environments altogether, such as multiple computers). The installation settings are captured in a *response file*, which serves as a template for the installation setup to be deployed. When you run the silent installation command for the target location, the response file is specified on the command line so that same installation settings are replicated by running the script.

To expedite the time for using an embedded solution, you can deploy a silent installation as a non-root installation, which does not require any system administration privileges or user and group **informix** accounts. For details, see information about non-root installation in the *IBM Informix Installation Guide for UNIX, Linux, and Mac OS X*.

Easy deployment

You can use the deployment tools to quickly and easily deploy IBM Informix to multiple locations.

You can take a snapshot of a fully configured Informix instance and its databases with the deployment assistant. You can then deploy that snapshot with the deployment utility to multiple target computers with one command.

The Informix embeddability tutorial provides step-by-step methods for using the deployment tools and writing a silent installation script. Specifically, the tutorial walks you through archiving of an installed database server instance with its associated data spaces on a template computer and deploying the snapshot on a different computer. A script-based method, such as described in the tutorial, can maximize the embeddability benefits of the deployment tools.

Automated administration

You can automate database server administration to minimize or eliminate the need for a database administrator after deployment of the database server instance as an embedded solution.

You can configure autonomic features that are included with IBM Informix:

- Autonomic configuration parameters control how the server does self-tuning and recovery. For example, the DYNAMIC_LOGS configuration parameter controls the dynamic allocation of log files when they are needed.
- Scheduler tasks perform maintenance tasks at configurable intervals for configurable thresholds. For example, the Auto Update Statistics (AUS) maintenance system updates stale table statistics.

You can monitor Informix:

- The **sysadmin:ph_alert** table contains server issues, ranked by severity. The issues are the results of built-in Scheduler sensors and event alarms.
- The IBM OpenAdmin Tool (OAT) for Informix shows the contents of the **sysadmin:ph_alert** table in graphical form and other performance indicators.

You can create customized responses and corrective actions to server issues:

- You can create Scheduler tasks to respond to alerts, event alarms, and other situations.
- You can use the SQL administration API in Scheduler tasks and other SQL statements to perform administration tasks remotely.

Maintain Informix availability

You can ensure that your applications can always access IBM Informix even if a single server fails by configuring a high-availability solution.

Sometimes Informix-embedded applications are deployed to environments where a DBA might not be immediately available. You can configure these applications to switch to another server if the original database server has a problem or is down for maintenance.

You can choose between different high-availability configurations depending on your needs:

- A high-availability cluster consists of a primary server and one or more secondary servers that are copies of the primary server. Secondary servers can be shared-disk, local, or remote. You can configure automatic connectivity based on available servers.

- Enterprise Replication allows a great deal of flexibility in how the participating servers are organized and whether all or a subset of the data is replicated.

Chapter 2. Deploying Informix software

The product supports embeddability tools that enable lightweight, faster deployment of the database server and related client programs.

Use the command-line deployment utility to deploy a snapshot of IBM Informix and create a new instance as a quick alternative to traditional installation, especially for deployments to many computers. You can configure the snapshot so that the instance is ready to meet your requirements immediately after deployment.

You can customize an RPM Package Manager image of an Informix installation and then distribute the image to other locations on supported Linux platforms.

The deployment utility

The IBM Informix deployment utility can deploy snapshots of pre-configured IBM Informix instances (with or without data) on one or more computers.

A snapshot is an image of an Informix database server that includes the installation directory, configuration settings, and any data spaces 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 customization of the snapshot.

Deploying a snapshot is quicker than installing Informix with the silent installation option. When you deploye a snapshot on a computer, you can replace the snapshot or remove it by using the deployment utility.

The following list describes some scenarios for which the deployment utility can be useful:

- You want to deploy a particular database server configuration on multiple computers. You can tune only one instance as the template instance, and then use the utility to deploy it on to other computers in silent mode.
- You want to clone an instance on the same computer or set up multiple instances quickly.
- You want to upgrade multiple instances to a different fix pack or version level to take advantage of newer product enhancements but this requires tuning certain configuration parameters or environment variables. You can tune the template instance, and then use the utility to upgrade other instances rapidly.
- You are embedding an Informix application on multiple computers and want to reduce application installation and setup time. You can specify the installation location and a single path for all application files including the database server files during deployment to avoid waiting for data loading and database server initialization.

Planning for deploying Informix

Before you deploy Informix, you should decide what you want to include in your deployment and how to configure the target server.

On the source server, you create a snapshot of the database server and, optionally, of the data in the database. Before you create a snapshot, decide on the properties of the snapshot. When you run the **ifxdeployassistant** utility in GUI mode, you can choose the following properties of the snapshot:

- The server features and components to include. You can choose which features and components to include in the snapshot. By default, all features and components are included in the snapshot.
- The data to include. You can create snapshots of your dbspaces. When you deploy them to the target computer, the dbspaces are already initialized. By default, no dbspaces are included in the snapshot.
- The archive format. You can choose one of three tar archive formats with or without compression, or a zip archive with compression. By default, the format is a tar archive with Gzip compression.

On the target server, you deploy the snapshot of the server and any dbspaces and configure the environment. Before you deploy the snapshot, decide on the properties of the target server. You specify the properties of the target server in the `ifxdeploy.conf` file and then run the **ifxdeploy** command with the `-config` option:

- Where to put the data. If you create snapshots of the data, you must create directories for the dbspaces, which can be the same or different from the directories on the source server. Create the directories with the proper ownership and permissions, copy the data snapshot files into the directories, and extract the files. By default, the **ifxdeploy** utility uses the same directories as on the source server. Specify different directories in the `ifxdeploy.conf` file.
- Where to put the server. You must choose an installation directory for the database sever. Create the directory and copy the server snapshot, the **ifxdeploy** utility, and the `ifxdeploy.conf` file into the directory.
- The configuration of the environment. You must set the INFORMIXDIR, INFORMIXSERVER, and INFORMIXPASSWORD parameters in the `ifxdeploy.conf` file. You can set other server properties.
- The configuration of the server. You can customize the configuration parameters for the target server. By default, an `onconfig` file is created based on the `onconfig.std` file. You can specify a different `onconfig` file or customize configuration parameters settings in the `ifxdeploy.conf` file.

Creating a snapshot with the deployment assistant

Use the deployment assistant to create a snapshot of a running Informix database server instance without shutting it down. You can reduce the snapshot footprint by selecting which features in the template instance to omit. You can include or exclude associated data spaces.

Before you create a snapshot, you must meet the following prerequisites:

- Java Runtime Environment (JRE) version 1.6 or higher must be on the system and must be in the PATH setting. The JRE must match the bit level of the Informix installation. You can use the JRE bundled with the Informix installation:`$INFORMIXDIR/extend/krakatoa/jre` on UNIX and Linux and `%INFORMIXDIR%\extend\krakatoa\jre` on Windows.
- You are able to run the deployment assistant on the computer that hosts the Informix instance to be used for the snapshot. (The deployment assistant cannot capture a snapshot from a remote computer.)
- You must have the CONNECT privilege on the **sysadmin** database to include any data spaces in the snapshot.

- Data spaces that you want to include in the snapshot must be stored in chunks that use buffered files (also known as cooked files).
- There must be sufficient disk space to save the snapshot in a file. You can store the snapshot in a compressed archive file.

The deployment assistant has two modes: the GUI mode and the command-line mode. You can only omit components and features by using the GUI mode.

To create a snapshot with the deployment assistant in GUI mode:

1. Run the **ifxdeployassist** command without options from the `%INFORMIXDIR%\bin` (Windows) or `$INFORMIXDIR/bin` (UNIX or Linux) directory to start the deployment assistant GUI.
2. If the deployment assistant does not detect the instance from which you want to create a snapshot, enter the connection information that pertains to the instance that you want to use as a template.
3. Specify the path, file name, and archive format for the snapshot that you are creating.
4. Select components and features that you want to include in or exclude from the snapshot.

   Important: The deployment assistant does not enforce functional interdependencies between product components and features.
5. If there are data spaces associated with the instance, select whether or not to include the data spaces in the snapshot. The deployment assistant GUI displays each data space, along with the chunk locations in each data space.
6. Confirm that the snapshot settings are correct.

To create a snapshot with the deployment assistant from the command line:

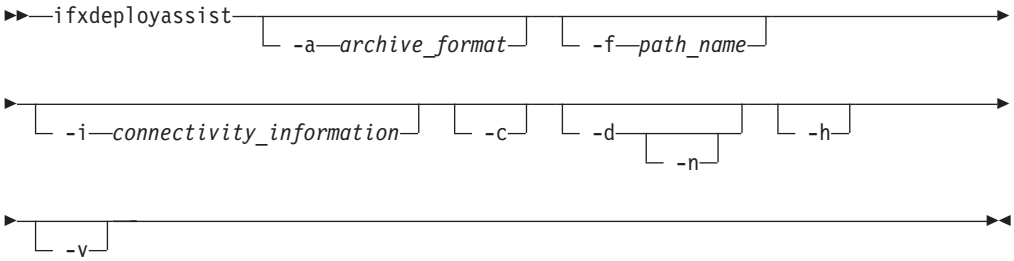
Run the **ifxdeployassist -c** command along with any other appropriate command options.

If data spaces are included in the deployed snapshot, the deployment assistant summary displays the full path name of the data spaces package.

The ifxdeployassist command

Use the **ifxdeployassist** command to create and customize a snapshot. If you do not pass any options on the command line, the deployment assistant runs in GUI mode.

Syntax



Command options

Table 2-1. Options for the **ifxdeployassist** *command*

Option	Purpose	Key Considerations
`-a archive_format`	Specifies the archive format and compression type to save the snapshot. The following values can be used in the place of *archive_format*: • `BZIP2` (tar archive with BZip2 compression) • `GZIP` (tar archive with GZIP compression) • `TAR` (tar archive with no compression) • `ZIP` (zip archive with compression)	If you do not pass the `-a` option in the command, the snapshot is saved as a tar archive with Gzip compression. If you use the standard Gzip format to create snapshots of data spaces, each data chunk must be less than or equal to 4 GB.
`-f path_name`	Specifies the path and file name of the snapshot to be created.	If you do not specify the `-f` option, the default file name of the server snapshot is: *server_name_yymmdd-hhmm*`.tar.gz` The *server_name* is the name of the server. The *yymmdd-hhmm* is the time the snapshot is taken. The default file name of the data snapshot is: *server_name_yymmdd-hhmm*`_db.tar.gz` The archive and compression selection set by the value in the `-a` option overrides any archive and compression settings indicated in the file name suffix set with the `-f` option. For example, if a command contains the following:`ifxdeployassist -a TAR -f C:\Informix\server.zip` then the snapshot will be saved as a tar archive with no compression rather than as a zip archive with compression.

Table 2-1. Options for the **ifxdeployassist** *command (continued)*

Option	Purpose	Key Considerations
`-i connectivity_information`	Specifies the connectivity information for the template instance from which to create a snapshot. If you do not use the `-i` option, the deployment assistant detects the connectivity information from the current environment and uses the same connectivity settings. In the place of *connectivity_information*, indicate specific connectivity settings for the snapshot to use after it is deployed. Set values for the following variables, separated by colons without spaces: • name of the database server instance • name of the template computer for which the database server instance is configured • service port number • protocol for the connection	The only valid values for the protocol information are `SQLI` and `DRDA`. If no protocol is specified, the command sets the connection to SQLI. The connectivity information here is contained in the `sqlhosts` file on UNIX or the SQLHOSTS registry key on Windows. **Example:** `ifxdeployassist -i hrapps:oscar:40:DRDA`
`-c`	Directs the deployment assistant to run in a command-line interface.	The `-c` option is required only if you are taking a snapshot of an instance that you do not want to customize (that is, accept all features of the template instance), but want to capture the snapshot in a scripting environment.
`-d`	Includes the data associated with the instance in the snapshot.	
`-n`	Captures a snapshot without the IBM Informix database server instance installation.	You must run the `-d` option in the command in order to use the `-n` option.
`-h`	Displays the online help for the **ifxdeployassist** command.	
`-v`	Runs the **ifxdeployassist** command in verbose mode.	

Usage

If you run the **ifxdeployassist** command without any options, the deployment assistant starts in GUI mode instead of as a command-line interface.

Examples of script-based usage of the deployment assistant

The examples of snapshot creation and modification below are based on a Windows environment in which:

- An installation of an Informix database server exists in `C:\informix`.
- The **ifxdeployassist** command is run in `C:\informix\bin`.
- The **INFORMIXSERVER** environment variable is set to informix1.

- The template instance has data associated with it.

Snapshot containing only the data space for the server

The following command creates a snapshot of the server only at `C:\informix\bin\informix1.tar.gz`.

```
ifxdeployassist -c
```

Snapshot containing the server and data, with specified archive type and full path name, from a template instance outside the current environment variable setting

The following command creates a snapshot of the database server at `C:\temp\snapshot_1.zip` and of the associated data at `C:\temp\snapshot_1_db.zip`. The value `informix2:localhost:9090` directs the deployment assistant to take a snapshot of the informix2 instance, which is configured for localhost, and to set the service port to 9090. Because neither DRDA® nor SQLI is specified after 9090, the connectivity port of the snapshot is set to use the SQLI protocol.

```
ifxdeployassist -c -d -i informix2:localhost:9090 -a zip -f C:\temp\snapshot_1
```

Snapshot in specified archive type capturing only the data associated with the instance, running in verbose mode

The following command creates a snapshot of data at `C:\Informix\bin\informix2_db.tar`. The template instance, host computer name, and connectivity information used are the same as described in the previous example.

```
ifxdeployassist -cvndi informix2:localhost:9090 -a tar
```

Creating a snapshot for deployment manually

Create a snapshot of IBM Informix that you can use with the deployment utility to place pre-configured instances on multiple computers.

Before you create a snapshot, you must meet the following prerequisites:

- **Windows:** Windows administrator privileges on the computer.
- **Linux and UNIX:** User **informix** or root privileges on the computer.
- Sufficient disk space to save the snapshot in a file. If space is limited, you might want to store the snapshot in a compressed file.

To create a snapshot, complete the following steps on the computer where you installed the Informix instance:

1. Shut down the instance in a consistent state with **onmode -kuy**.
2. Create a snapshot of the following items:
   - Informix installation directory

     Tip: Ensure you include the `IDSFILES.txt` file, which is in the installation directory. That file is required if you want to use the deployment utility to remove the snapshot from the target computer after you deploy it.
   - Configuration settings
   - Optional: Data spaces associated with the instance

     Tip: Store the components of the snapshot in a compressed file if you want to save space. If you compress the snapshot in a BZIP2, GZIP, TAR, or ZIP

format, you can use the `-file` option with the deployment utility to extract the snapshot instead of specifying a customized decompression command.

3. Optional: After you create the snapshot, you can restart the instance.

Deploying a snapshot with the deployment utility

Deploy a snapshot of an instance by using the **ifxdeploy** utility on the target computer.

Before you deploy a snapshot, you must meet the following prerequisites:

- **Windows:** Windows Administrator privileges on the target computer.
- **Windows:** The required Visual C++ runtime libraries must exist on the target computer. You can install these libraries by one of the following methods:
  - Copy the `%INFORMIXDIR%\bin\vcredist.exe` file to the target computer and run the executable.
  - Run the following command on the target computer:

    ```
    vcredist.exe /q:a /c:"msiexec /i vcredist.msi /qn"
    ```
- **Linux: and UNIX:** Root privileges on the target computer.
- The target computer has sufficient disk space for the snapshot. You must have the same amount of space as was used on the template computer. The space required depends on what you included in the snapshot, such as data, extra files, or other applications.
- The target computer must be in the same operating system family as the source computer.

This procedure describes how to deploy that snapshot with a configuration file. Alternatively, you can use command-line options to specify the same information that is contained in the configuration file.

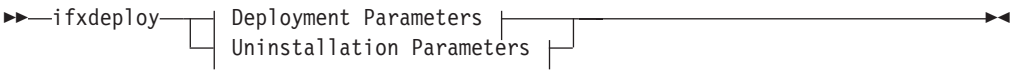
To deploy the snapshot on the target computer with a configuration file:

1. Create the directory for the server and save the following files in that directory:
   - The server snapshot
   - The **ifxdeploy** utility file
   - The `ifxdeploy.conf` file
2. If you have a data snapshot, create a directory for the data, save the data snapshot in that directory, and uncompress the data spaces. Make the root dbspace directory the same absolute path as the root dbspace on the source computer, unless you plan to set the ROOTPATH parameter in the `ifxdeploy.conf` file. If the data spaces on the source computer are not all in the same directory, set the RELOCATE parameter in the `ifxdeploy.conf` file to the correct directory names for each data space.
3. Customize the `ifxdeploy.conf` file for the target environment. Set the following parameters and any other parameters you need:
   - INFORMIXDIR
   - INFORMIXSERVER
   - INFORMIXPASSWORD
4. Run the **ifxdeploy** command with the `-config=ifxdeploy.conf` option and any other appropriate options.

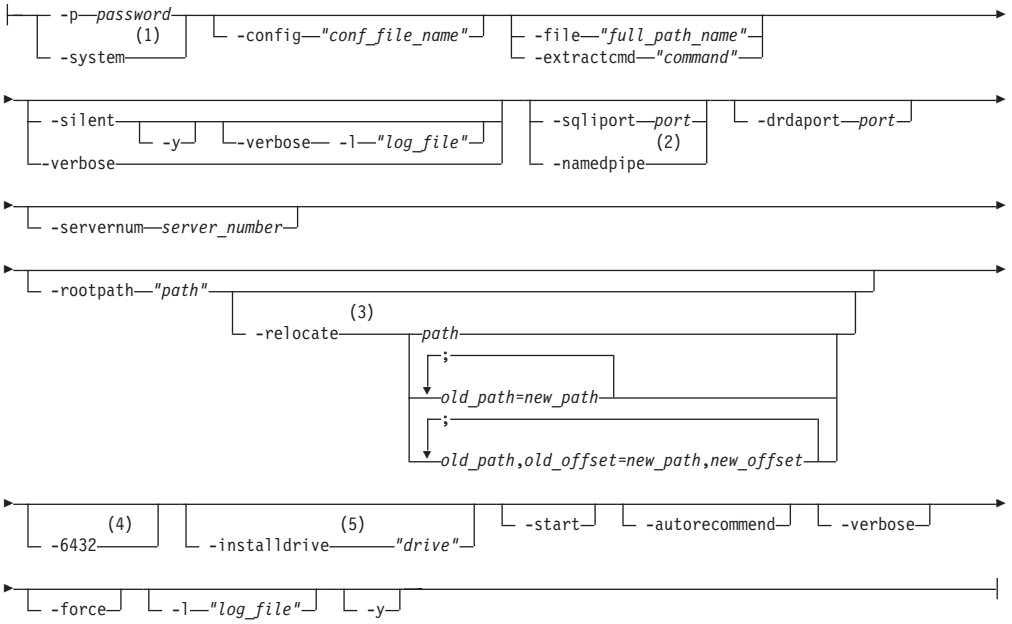
The ifxdeploy command: The deployment utility

Use the **ifxdeploy** command to deploy a snapshot or remove a snapshot that you already deployed.

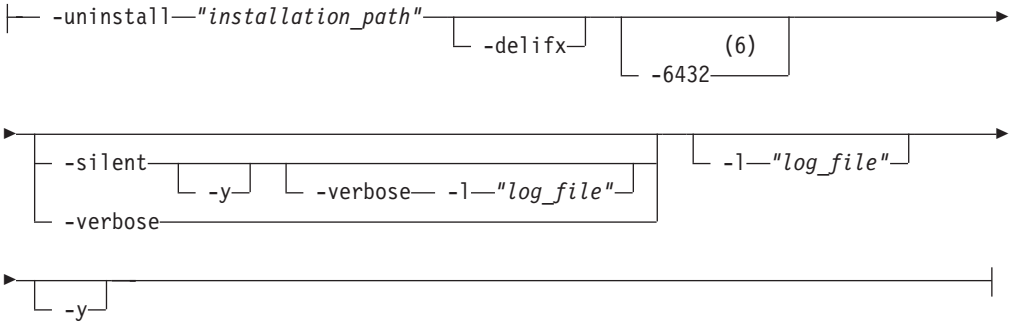
Syntax



Deployment parameters:



Uninstallation parameters:



Notes:

1 **-system** functions on Windows only

2 **-namedpipe** functions on Windows only

3 **-relocate** can run without **-rootpath** option if ROOTPATH is set in an ONCONFIG parameter

4 **-6432** functions on Windows only

5 **-installdrive** functions on Windows only

6 **-6432** functions on Windows only

Command options

Table 2-2. The **ifxdeploy** *command options*

Element	Purpose	Key considerations
`-autorecommend`	Calculates the optimal settings for some `onconfig` file parameters. This option generates an alternative customized configuration file.	The alternative configuration file might have suggested changes for certain configuration parameters that operate when the server is restarted. **Windows:** The name of the alternative configuration file is `%ONCONFIG%.autorec`. **Linux and UNIX:** The name of the alternative configuration file is `$ONCONFIG.autorec`.
`-config "conf_file_name"`	Specifies deployment configuration file to run with utility.	
`-delifx`	**Windows:** Removes user **informix** and the **Informix-Admin** group. **Linux and UNIX:** Removes user **informix** and group **informix**.	The `-delifx` option functions when there is only one IBM Informix installation on the computer.
`-drdaport port`	Specifies the Distributed Relational Database Architecture™ (DRDA) service port for use with the IBM Common Clients.	The default is 9089.
`-file "full_path_name"`	Extracts a compressed snapshot *only if* the file specified in *"full_path_name"* is in BZIP2, GZIP, TAR, or ZIP format.	**Windows:** Indicate the location of the compressed image as an absolute path. Example: `-file C:\IBM\informix\my_archive.zip` **Linux and UNIX:** Example: `-file /opt/IBM/informix/ifxdir.tgz` If the snapshot is not in the BZIP2, GZIP, TAR, or ZIP compression format, use the `-extractcmd` option instead.
`-extractcmd "command"`	Extracts a compressed snapshot, regardless of compression format. This option primarily is intended for when you want to extract a snapshot that is *not* in BZIP2, GZIP, TAR, or ZIP format or when you have specialized extraction procedure.	Specify the command and the file that contains the snapshot files. If the snapshot is in BZIP2, GZIP, TAR, or ZIP compression format, then use the `-file` option instead. Example on Windows:`-extractcmd "tar -xf C:\my_archive.tar c:\informix"` Example on Linux and UNIX:`-extractcmd " ( cd /placetoextract ; myextract /opt/IBM/informix/ifxdir) "`

Table 2-2. The **ifxdeploy** *command options (continued)*

Element	Purpose	Key considerations
`-force`	Overwrites the existing instance settings on the target computer.	
`-installdrive "drive"`	This option is for Windows only. Specifies the drive for the directory containing data spaces (typically, this is the `IFMXDATA` directory).	The default drive is `C:`. You must have enough space for the Informix installation and databases.
`-l "file_path"`	Sends status messages to a text file.	Indicate the full path and file name for the log file that you want to use. If the file does not exist, it will be created for you.
`-namedpipe`	This option is for Windows only. Sets the default database server network protocol to a named pipe connection.	If you specify both `-namedpipe` and `-sqliport` options, the deployment utility uses only one of them.
`-p password`	Specifies the user **informix** password used to create the Informix service.	Specify the password for user **informix** on the target computer. If you specify a password and the user **informix** does not exist on the target computer, the user will be created and will have the specified password.
`-relocate path`	Specifies new parent directory of chunks in the deployed instance. You must have the appropriate ownership and permissions to modify the directories.	You can specify a single new parent directory or map multiple separate chunk paths to different locations. In addition, you can substitute offsets of the old paths with new values if you are indicating multiple new parent directories. **Windows:** If the parent directories have spaces in the paths, place the string of paths after the `-relocate` option in double quotation marks. Example:`ifxdeploy -relocate "C:\Program Files\IBM\IBM Informix\11.50\ tmp=D:\IFMXDATA\myserver"` **Linux and UNIX:** If multiple parent directories are specified and have spaces in the paths, place the string of paths after the `-relocate` option in single quotation marks. Example:`ifxdeploy -relocate '/opt/IBM/informix1150/ tmp=/local/data/myserver; /opt/IBM/informix1150/ dbspace=/local2/data/myserver'`

Table 2-2. The **ifxdeploy** *command options (continued)*

Element	Purpose	Key considerations
`-rootpath "path"`	Indicates the location of the root dbspace.	**Windows:** Default path:*drive:*`\ifmxdata\`*server_name*`\rootdbs_dat.000` **Linux and UNIX:** Default path:`$INFORMIXDIR/`*server_name*`/rootdbs`
`-servernum` *server_number*	Specifies the server number of the instance.	The number must be an integer from 0 to 255. The default is 0.
`-silent`	Directs the utility to run in silent mode.	When you use this option, messages do not display on the screen but are written in a log file. If you use the `-silent` option and `-verbose` option together, you must also include the `-l` argument.
`-sqliport` *port*	Specifies the SQLHOSTS service port for the server instance.	The default port is 9088. If you specify both `-namedpipe` and `-sqliport`, the deployment utility uses only one of them.
`-start`	Sets the number of seconds for the deployed instance to wait for automatic initialization of Informix shared memory to bring the database server online before returning to the command line.	You must change the setting from zero to a workable number for the `-start` option to function. If you do not pass a value with the `-start` option, the default number of seconds is 600. This option supports the deployment and starting of an instance in a single operation. Using this option, an online operational database is silently deployable.
`-system`	This option is for Windows only. Creates an Informix service that logs on to the operating system as system user.	
`-uninstall` *installation_path*	Removes a snapshot that was originally deployed by the deployment utility.	
`-verbose`	Runs the command in verbose mode.	
`-y`	Runs the command without prompting for confirmation.	
`-6432`	This option is for Windows only. Redirects registry access to 32-bit registry view.	Use this option if you are deploying a 32-bit Informix instance on a 64-bit computer.

Usage

Before you use this command, create a snapshot and copy it to the computer where you want to deploy the snapshot. See "Creating a snapshot for deployment manually" on page 2-6 for more information.

Run this command, with options, on the computer where you want to deploy the snapshot. When you specify a value for a parameter, if the value contains a space, enclose the value in double quotation marks. You must run the command either as an Administrator user or as the **root** user.

You must provide the location of the snapshot by using the `-extractcmd` option, the `-file` option, or by setting the SNAPSHOT parameter in the `ifxdeploy.conf` file.

If the snapshot is in a different compressed format, you must extract it. Use the `-file` option to extract a snapshot if it is in BZIP2, GZIP, TAR, or ZIP compression format. If the snapshot is in a different compression format, use the `-extractcmd` option with a customized command or script.

If you included data spaces in your snapshot in compressed format, you must extract the data spaces before you run the **ifxdeploy** command. Alternatively, you can write a shell script to use with the `-extractcmd` option that specifies how to extract both the server snapshot and the data spaces snapshot.

Use the `-force` parameter to deploy a snapshot to multiple computers and override the database server instance settings on the target computer.

On Windows, the **ifxdeploy** command deploys a snapshot of Informix, which is not visible. If you do not want to use Informix, use the **ifxdeploy** utility to uninstall the it.

Examples of deployment utility usage on Windows

Deploying an Informix instance in silent mode

In this example, the user **informix** is created on the target server and has the password `mypassw0rd`. The command will run in silent mode, without prompting for confirmation. The snapshot is extracted from the `my_archive.zip` file by 7-zip software if it is installed on the operating system.

```
ifxdeploy.exe -p mypassw0rd -y -silent -file C:\IBM\informix\my_archive.zip
```

Deploying an Informix instance as local system user and specifying an extraction method

In this example, the snapshot is extracted from the `C:\my_archive.tar` file on the target computer.

```
ifxdeploy.exe -system -extractcmd "tar -xf C:\my_archive.tar c:\informix"
```

Because the `tar -xf` part of the argument to the `-extractcmd` option contains a space, the argument and the paths after it are enclosed in double quotation marks.

Creating a new server instance

In this example, the SQLHOSTS port number is 9090, the server number for the instance is 2, and a full path is specified for the log file.

```
ifxdeploy.exe -silent -y -sqliport 9090 -servernum 2 -l C:\my_log.txt
```

Creating an instance as local system user with DRDA enabled

In this example, the SQLHOSTS port is 9090, the DRDA port is 9096, and the command will run in verbose mode. The snapshot used here is not in a compressed file format, so neither the `-file` option nor the `-extractcmd` option are required.

```
ifxdeploy.exe -system -y -sqliport 9090 -drdaport 9096 -verbose
```

Dynamic relocation of chunks to single parent directory

In this example, the chunks are relocated to a single parent directory and indicate location of the root dbspace:

```
ifxdeploy -rootpath D:\IFMX\ex1\rootdbs.001 -relocate D:\IFMX\ex2
```

Chunk relocation to multiple paths

In this example, the location of the root dbspace has been specified with the ROOTPATH configuration parameter, so the command is not required to run with the `-rootpath` option. Each mapping between the old path and the new path is separated by a semicolon.

```
ifxdeploy -relocate C:\IFMXDATA=D:\IFMXDATA;C:\IFMXLOGSPACE=E:\IFMXLOGSPACE
```

Chunk relocation to multiple paths with new offsets

In this example, the location of the root dbspace has been specified with the ROOTPATH configuration parameter, so the command is not required to run with the `-rootpath` option. The mapping of the old paths and the new paths includes offset values (in KB), which are indicated after the commas.

```
ifxdeploy -relocate C:\IFMXDATA,0=D:\IFMXDATA,4;C:\IFMXDATAB,3=D:\IFMXDATAB,5
```

Automatic startup and initialization of the deployed instance

In this example, the command runs as follows:

- Does not prompt for confirmation (the `-y` option).
- Extracts a snapshot located at `C:\work\ifxdir.zip`.
- Sets the server number of the deployed instance to 2 (the `-servernum` option).
- Directs the deployed instance to attempt automatic initialization and startup (the `-start` option) within 300 seconds. If initialization is not complete after 300 seconds, the computer returns to the command line.

```
ifxdeploy -file C:\work\ifxdir.zip -servernum 2 -start 300 -y
```

Automatic startup, initialization of the deployed instance, and creation of a customized configuration file

In this example, the command runs as follows:

- Does not prompt for confirmation (the `-y` option).
- Extracts a snapshot located at `C:\work\ifxdir.zip`.
- Sets the server number of the deployed instance to 2 (the `-servernum` option).
- Directs the deployed instance to attempt automatic initialization and startup (the `-start` option) within 300 seconds. If initialization is not complete after 300 seconds, the computer returns to the command line.
- Creates an alternative configuration file with optimal settings.

```
ifxdeploy -file C:\work\ifxdir.zip -servernum 2 -start 300 -autorecommend -y
```

Automatic startup and initialization of the deployed instance with chunk relocation to multiple paths

In this example, the command runs as follows:

- Runs in verbose mode (the `-verbose` option) and does not prompt for confirmation (the `-y` option).
- Extracts a snapshot located at `C:\work\ifxdir.zip`.
- Directs the deployment utility to dynamically relocate chunks to a single parent directory (the `-relocate` option) and sets the location of the root dbspace (the `-rootpath` option).
- Sets the server number of the deployed instance to 2 (the `-servernum` option).
- Directs the deployed instance to attempt automatic initialization and startup (the `-start` option). Because no value is set with the `-start` option, the computer waits the default time of 600 seconds before returning to the command line.

```
ifxdeploy -file C:\work\ifxdir.zip -verbose -servernum 2
-relocate C:\work\chunks\instance2
-rootpath C:\work\chunks\instance2\rootdbs -start -y
```

Examples of deployment utility usage on Linux and UNIX

Extracting a snapshot saved as a `.tgz` file with Gzip and other options

In this example, the deployment utility extracts a snapshot saved as `/opt/IBM/informix/ifxdir.tgz` by using Gzip. The command will run in verbose mode, create the log file `/tmp/mylog`, and not prompt for confirmation.

```
ifxdeploy -file /opt/IBM/informix/ifxdir.tgz -l /tmp/mylog -verbose -y
```

Specifying SQLHOSTS settings and the server number for the deployed instance

In this example, the Informix files are already in place and you are creating a new instance by specifying port numbers for DRDA and the service port, as well as setting the server number for the instance. The snapshot is not compressed, so neither the `-extractcmd` option nor the `-file` option are required. The **INFORMIXDIR** and **INFORMIXSERVER** environment variables have been updated to new values as necessary.

```
ifxdeploy -sqliport 9093 -drdaport 9094 -servernum 3
```

Dynamic relocation of chunks to single parent directory

In this example, the chunks are relocated to a single parent directory and indicate location of the root dbspace:

```
ifxdeploy -rootpath /opt/ibm/IDS/exa/space -relocate /opt/ibm/IDS/exa/space2
```

Chunk relocation to multiple paths

In this example, the location of the root dbspace has been specified with the ROOTPATH configuration parameter, so the command is not required to run with the `-rootpath` option. Each mapping between the old path and the new path is separated by a semicolon.

```
ifxdeploy -relocate /opt/IBM/ex3=/idsb/myserver;/opt/IBM/ex4=/idslogs/myserver
```

Chunk relocation to multiple paths with new offsets

In this example, the location of the root dbspace has been specified with the ROOTPATH configuration parameter, so the command is not required to run with the -rootpath option. The mapping of the old paths and the new paths includes offset values (in KB), which are indicated after the commas.

```
ifxdeploy -relocate /opt/IBM/dbspaces,0=/lv1/data,4;/opt/IBM,2=/ids/myserver,6
```

Automatic startup and initialization of the deployed instance

In this example, the command runs as follows:

- Does not prompt for confirmation (the -y option).
- Extracts a snapshot located at /work/ifxdir.tgz (the -file option).
- Sets the server number of the deployed instance to 2 (the -servernum option).
- Directs the deployed instance to attempt automatic initialization and startup (the -start option) within 300 seconds. If initialization is not complete after 300 seconds, the computer returns to the command line.

```
ifxdeploy -file /work/ifxdir.tgz -servernum 2 -start 300 -y
```

Automatic startup, initialization of the deployed instance, and creation of a customized configuration file

In this example, the command runs as follows:

- Does not prompt for confirmation (the -y option).
- Extracts a snapshot located at /work/ifxdir.tgz (the -file option).
- Sets the server number of the deployed instance to 2 (the -servernum option).
- Directs the deployed instance to attempt automatic initialization and startup (the -start option) within 300 seconds. If initialization is not complete after 300 seconds, the computer returns to the command line.
- Creates an alternative configuration file with optimal settings.

```
ifxdeploy -file /work/ifxdir.tgz -servernum 2 -start 300 -autorecommend -y
```

Automatic startup and initialization of the deployed instance with chunk relocation to multiple paths

In this example, the command runs as follows:

- Runs in verbose mode (the -verbose option) and does not prompt for confirmation (the -y option).
- Extracts a snapshot located at /work/ifxdir.tgz.
- Directs the deployment utility to dynamically relocate chunks to a single parent directory (the -relocate option) and sets the location of the root dbspace (-rootpath option).
- Sets the server number of the deployed instance to 2 (the -servernum option).
- Directs the deployed instance to attempt automatic initialization and startup (the -start option). Because no value is set with the -start option, the computer waits the default time of 600 seconds before returning to the command line.

```
ifxdeploy -file /work/ifxdir.tgz -verbose -servernum 2
-relocate /work/chunks/instance2
-rootpath /work/chunks/instance2/rootdbs -start -y
```

The ifxdeploy.conf file: The deployment utility configuration file

The ifxdeploy.conf file is a text-file template in which you can configure an instance snapshot before deploying it with the **ifxdeploy** command.

Purpose

The ifxdeploy.conf file is in $INFORMIXDIR/etc/ on UNIX and Linux and %INFORMIXDIR%\etc on Windows.

The ifxdeploy.conf file allows for more dynamic customization than the command-line options because in the file you can specify any configuration parameter values, set key environment variables, and create multiple database server aliases. You can save and reuse the file to deploy instances to other locations. The file supports the same functionality as the **ifxdeploy** command options, but also additional functionality that is useful for embedding IBM Informix when you are deploying a snapshot in multiple locations that require minimal or no modification in instance setup. See "Planning for deploying Informix" on page 2-1 for a description of the major embeddability features of the ifxdeploy.conf file.

Guidelines for Usage

To apply configuration settings of the ifxdeploy.conf file to an instance you are deploying, enter the file name as an argument to the -config option when you run the **ifxdeploy** command.

If you set different values for the same instance in the ifxdeploy.conf file and as an option to the **ifxdeploy** command option, the deployment utility uses the value specified on the command line. For example, if you specify -p mypassword1 on the command line but set the INFORMIXPASSWORD parameter to mypassword2 in the ifxdeploy.conf file, the deployed instance requires mypassword1 for authentication.

Parameters

The ifxdeploy.conf file is value pair based. If there is a parameter with a default value that you want to change, provide the value in an uncommented line.

Attention: The values of parameters that are set on the **ifxdeploy** command line overwrite the values of the same parameters in the ifxdeploy.conf file.

The following table explains the parameters in the same order that they appear in the configuration template file.

Table 2-3. The ifxdeploy.conf file parameters

Parameter	Description	Example of value setting (uncommented line)
INFORMIXSERVER	Primary database server name. Must be set either here or as environment variable before deployment (no default value is provided).	INFORMIXSERVER deploy3

Table 2-3. The ifxdeploy.conf file parameters (continued)

Parameter	Description	Example of value setting (uncommented line)
PROTOCOL1	Primary network protocol. **Linux and UNIX**: This is equivalent to the sqlhosts file's NETTYPE setting. **Windows:** This is equivalent to the PROTOCOL field of the SQLHOSTS registry.	PROTOCOL1 olscoctcp
SQLIPORT	SQLHOSTS service port for the server instance (not required for onipcnmp). The range of permissible values is from 1 to 65536.	SQLIPORT 9088
DRDAPORT	The Distributed Relational Database Architecture (DRDA) service port for use with the IBM Common Clients. The range of permissible values is from 1 to 65536.	DRDAPORT 9089
SERVERNUM	The server number. Corresponds to the SERVERNUM configuration parameter. The range of permissible values is from 0 to 255.	SERVERNUM 100
INFORMIXSQLHOSTS	**Linux and UNIX**: Full path to sqlhosts file for the instance to use. **Windows:** Pointer to remote computer containing SQLHOSTS registry settings that the deployed instance is to use.	
BEGIN ALIAS . . . END ALIAS	Specifies new database server aliases and related SQLHOSTS connectivity settings for the deployed instance. The optional OPTIONS line sets an SQLHOSTS parameter value. In the example, b=32767 sets buffers.	BEGIN ALIAS SERVERNAME alias1 PROTOCOL drsoctcp PORT 9091 OPTIONS b=32767 END ALIAS
INFORMIXDIR	Path for the deployed instance. Must be set here or as environment variable (no default path is provided).	**Windows:** C:\tmp\informix **UNIX andLinux**: /tmp/informix
ONCONFIG	The onconfig file name. If none is specified here and no **ONCONFIG** environment variable is set, a new file is created from onconfig.std.	onconfig.sample

Table 2-3. The ifxdeploy.conf file parameters (continued)

Parameter	Description	Example of value setting (uncommented line)
START	Set this to the number of seconds for the deployed Informix instance to wait for database server initialization to complete before returning to the command line. Use this parameter in an embeddability environment that requires noninteractive startup of Informix. The setting of 0 directs the deployed Informix instance to not attempt automatic startup. If you do not specify a value with the `-start` option, the target computer waits the default time of 600 seconds before returning to the command line.	`START 90`
SNAPSHOT	This parameter can only be used if you are deploying an instance from a snapshot compressed as a `.tgz` file on UNIX or Linux or a `.zip` file on Windows. Indicates the location of a supported, compressed archive type containing the snapshot. This parameter is equivalent to the `-file` command-line option.	
RELOCATE	Set chunk paths for the deployed instance. You can indicate a parent directory for all chunk path names or map them individually to separate parent directories. You can also deploy the chunks with specific offset values. Offset values are in KB. You must have the appropriate ownership and permissions to update the directories.	*Example 1: Relocate all chunk paths to one directory:* Windows: `C:\example1` UNIX and Linux: `/example1` *Example 2: Select individual chunk paths and specify the directories where the paths are relocated:* Windows: `C:\ex2=C:\ex3; C:\ex4=C:\ex5` UNIX and Linux: `/ex2=/ex3;/ex4=C:ex5` *Example 3: Change multiple chunk paths and offsets:* Windows: `C:\ex6,10=C:\ex7,100; C:\ex8,20=C:\ex9,200` UNIX and Linux: `/ex6,10=/ex7,100; /ex8,20=/ex9,200`
INFORMIXPASSWORD	Password for user **informix** on the target computer. Sets password to what you enter as a value if no user **informix** exists on the computer. If you do not supply a password, you are prompted to enter a password by the **ifxdeploy** utility.	`INFORMIXPASSWORD` *password*

Table 2-3. The ifxdeploy.conf file parameters (continued)

Parameter	Description	Example of value setting (uncommented line)
SYSTEM This parameter is for Windows only.	Sets whether the deployed instance will log on to Windows as local system account. The default value is 0 (Informix logs on as user **informix**).	SYSTEM 1 (this directs instance to log on as local system account but user **informix** is created) SYSTEM 2 (this deploys instance without creation of user **informix**)
LOGFILE	Sets the full path name for the log file of the deployment utility's errors and messages.	**Windows:** LOGFILE c:\my_log.txt **UNIX andLinux**: LOGFILE /tmp/mylog
LOGLEVEL	Sets amount of information to write to log. See the ifxdeploy.conf file for permissible values.	LOGLEVEL 5
SILENT	Sets whether the utility displays console output while it is running.	SILENT 1 (no console output) SILENT 0 (displays console output)
FORCE	Overwrites existing environment variable and onconfig file settings of the target computer.	FORCE 0 (Does not overwrite) FORCE 1 (Overwrites existing settings)
INSTALLDRIVE This parameter is for Windows only.	Specifies the directory in which the deployed instance's dbspaces will be created on Windows.	INSTALLDRIVE C
ROOTPATH	Sets the location of the root dbspace.	**Windows:** Default path: *drive:*\ifmxdata*server_name*\rootdbs_dat.000 **UNIX and Linux** Default path: $INFORMIXDIR/*server_name*/rootdbs
WIN6432 This parameter is for Windows only.	Set this to 1 if installing 32-bit Informix on 64-bit Windows.	WIN6432 1
BEGIN ONCONFIG . . . END ONCONFIG	Specify values for any configuration file parameters. Enter each parameter and value exactly as they would be entered in the onconfig file. Can be used to overwrite specific onconfig file parameter values or instead of providing an onconfig file.	BEGIN ONCONFIG LOCKS 10000 END ONCONFIG

Table 2-3. The ifxdeploy.conf file parameters (continued)

Parameter	Description	Example of value setting (uncommented line)
BEGIN AUTORECOMMEND . . . END AUTORECOMMEND	Specifies the following parameters that are used to generate recommended values in the onconfig file: **MAXCPUS** Maximum number of processors and cores that the database server can use. Default: 1 Minimum value: 1 Maximum value: none **MAXDISK** Maximum amount of disk space that the database server can use. Default: 2048 MB Minimum value: 256 MB Maximum value: none **MAXMEM** Maximum amount of memory that the database server can use. Default: 512 MB **MAXUSERS** Maximum number of expected online transaction processing (OLTP) application users. Default: 32 **MAXDSUSERS** Maximum number of expected decision-support system (DSS) application users. Default: 4 **RTO_SERVER_RESTART** Recovery time objective for a server restart. Default: 60 seconds Minimum value: 60 seconds Maximum value: 1800 seconds	BEGIN AUTORECOMMEND MAXCPUS 1 MAXDISK 2048 MAXMEM 512 MAXUSERS 32 MAXDSUSERS 4 RTO_SERVER_RESTART 60 END AUTORECOMMEND
CLONE	Set this to 1 to deploy a clone of a source server. Specify the information for the source server in the parameters nested between the BEGIN CLONE and END CLONE statements of the ifxdeploy.conf file.	1
SOURCESERVER	Specifies the name of the source server.	SOURCESERVER clone 3

Table 2-3. The `ifxdeploy.conf` file parameters (continued)

Parameter	Description	Example of value setting (uncommented line)
SOURCEIPADDR	Specifies the IP address of the source server.	
SOURCEPORT	Specifies the port number of the source server's listener thread.	
CLONEIPADDR	Specifies the IP address of the clone server.	
DISPOSITION	Specifies the final disposition of the clone server.	`DISPOSITION RSS`
TARGETSIZE	Specifies the size of clone server.	`TARGETSIZE medium`
USELOCAL	Indicates to use local configuration after merging with source configuration.	`USELOCAL 1`
TRUSTED	Specifies that the user is trusted and that it is not necessary to obtain a user ID and password to access the server.	`TRUSTED 1`
USERNAME	Specifies the name of user for connecting to source server	`USERNAME informix`
PASSWORD	Specifies the password of the name specified in the USERNAME parameter.	`PASSWORD password`

Removing a snapshot with the deployment utility

Use the deployment utility to remove a snapshot that was originally deployed by the deployment utility. In addition, you can use this utility to remove user **informix** and the **Informix-Admin** group (Windows) or to remove user **informix** and group **informix** (Linux) from the operating system.

Windows: You must have Windows administrator privileges to remove the snapshot or remove the user and group objects.

Linux: You must have root privileges.

You must have the `IDSFILES.txt` file in the `etc` subdirectory of the installation path.

To remove a snapshot:

Run the **ifxdeploy** command with the uninstallation options on the computer where the snapshot is deployed. See "The ifxdeploy command: The deployment utility" on page 2-8 for details about the command syntax.

The following command is an example of how to use the deployment utility to remove a snapshot and to display runtime status messages:

Windows:

```
ifxdeploy.exe -u C:\tmp\informix -verbose
```

Linux:

```
ifxdeploy -u /tmp/informix -verbose
```

The following is an example of how to uninstall a snapshot and to remove user **informix** and the administrative group:

Windows:

```
ifxdeploy.exe -u C:\tmp\informix -verbose -delifx
```

Linux:

```
ifxdeploy -u /tmp/informix -verbose -delifx
```

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

Create an RPM Package Manager image of a customized IBM Informix product installation, which you can then use to deploy with silent installation.

See the Machine Notes for the specific Linux operating systems that are supported. RPM-based installation and deployment is not available on other operating systems. Before running the IBM Informix installation media with RPM:

- Check that RPM is installed on your computer.
- Log in to the computer as root user if you want to perform a standard installation that will run as privileged user **informix**.
- Ensure that user **informix** and group **informix** exist on your system if you want to perform a standard installation that will run with root-level privileges.

Perform the following task if you want to create an RPM image for redistribution. Creation and customization of the image does not create a functional installation, but the task can save time in the overall beginning-to-end process of Informix software deployment and embedded application development when you need a single installation setup repeated in multiple environments.

When creating an RPM image, you can only complete the installation application in custom setup. However, the installation application for RPM imaging does not support the option to create an instance of the database server automatically after the installation application ends. Typical setup also is not supported.

1. Follow the instructions for an Informix software bundle installation with selected features (custom setup) in the *IBM Informix Installation Guide for UNIX, Linux, and Mac OS X*.

   After you finish running the installation application, the rpm image is located under `$INFORMIXDIR/RPMS`. The image file name begins with `informix-` and ends with the suffix `.rpm`, but the remainder of the file name varies by the following factors:

   - full version identifier: either `11.70.UC2-0` (for version 11.70, fix pack 2 on 32-bit Linux) or `11.70.FC2-0` (for version 11.70, fix pack 2 on 64-bit Linux)
   - architecture: either `i386` (for 32-bit Linux x86) or `x86_64` (for 64-bit AMD)

   For example, an image made with Informix version 11.70.UC2, fix pack 2, for 32-bit Linux x86, has the file name `informix-11.70.UC2-0.i386.rpm`.
2. Log in with root privileges on the computer where you want to deploy the Informix installation. Root-level privileges are required to deploy the RPM image, even if you are deploying a non-root installation.

3. Copy the image to a local directory.
4. Run the following command:

   ```
   rpm -i image_file_name --prefix absolute_installation_path
   ```

   If you do not specify an installation path with the `--prefix` option, then the default installation location is `/opt/IBM/Informix`.

   Tip: Record the image file name after you have completed a deployment. You will need to use the image file name if you uninstall the product.

Attention: If you are deploying a non-root installation of the database server, then you must change the ownership of the installation directory on the operating system to the non-root owner.

To remove an installation with RPM:

You must be logged in with root-level privileges before completing the following steps:

Tip: Obtain the file name of the image used to create the installation to quickly refer to information that is needed for the uninstallation command.

1. Run the following command, replacing *release_identifier* with major Informix release number, minor release number, and operating system platform used for installation:

```
rpm -e informix-release_identifier-0
```

For example, if the installation that you want to remove was created using an image generated from Informix Version 11.70, fix pack 2 on 32-bit Linux, you would run the following command:

```
rpm -e informix-11.70.UC2-0
```

Chapter 3. Informix embeddability deployment tutorial

The purpose of this tutorial is to provide steps and scripts needed for silent end-to-end deployment of IBM Informix on Linux and Windows using the deployment assistant and the deployment utility.

The IBM Informix Embeddability Toolkit is a logical collection of the following components:

- **ifxdeployassist**: the deployment assistant (DA).
- **ifxdeploy**: the deployment utility (DU).
- `ifxdeploy.conf`: the DU's configuration file.
- **ifx_silent_deploy**: an example script that automates silent deployment using DU. The Linux shell script and the Windows batch script for this silent deployment example are posted separately on the Technote at http://www.ibm.com/support/docview.wss?uid=swg21446737.

The following tasks are covered in this tutorial for silent deployment of Informix:

- Create a Snapshot for Deployment: Using the DA, archive an installed Informix server instance and its dbspaces on the template computer for future deployments.
- Silently Deploy Informix from the Snapshot: Using the DU, its configuration file, and **ifx_silent_deploy**, silently deploy a copy of the archived Informix server instance and its dbspaces on a target computer.

Introducing the Informix deployment assistant

The deployment assistant is used to create snapshots of IBM Informix instances and their associated dbspaces, which can be later deployed on multiple target computers using the deployment utility.

The deployment assistant also allows users to reduce the footprint of an Informix installation by displaying file associations at the time of packaging.

Using the deployment assistant

The deployment assistant is included on the Informix server at `$INFORMIXDIR/bin` on Linux and at `%INFORMIXDIR%\bin` on Windows. To use the deployment assistant, you must run the following command from a command line:

```
ifxdeployassist
```

Introducing the Informix deployment utility

You can use the deployment utility to deploy snapshots of pre-configured instances and related dbspaces onto target computers.

You must install IBM Informix before creating a snapshot. You can choose to include a fully configured server instance in the snapshot or create the instance at the time of deployment. The advantage of using the deployment utility is that you do not need to install and configure Informix on each target computer. You simply

deploy a previously installed and fully configured instance. This gives you the ability to quickly recreate a fully configured instance on one or more target computers.

Using the deployment utility

The **ifxdeploy** utility is included with the Informix server at `$INFORMIXDIR/bin` on Linux and UNIX and at `%INFORMIXDIR%\bin` on Windows. To use the deployment utility, you must run the **ifxdeploy** utility either from a command line or as part of an application integration script. You can run the utility in silent mode without any user interaction.

The ifxdeploy utility

The **ifxdeploy** utility is located in `$INFORMIXDIR/bin` on Linux and UNIX and at `%INFORMIXDIR%\bin` on Windows.

To use the deployment utility, you must run the **ifxdeploy** utility either from a command line or as part of an application integration script. You can run the utility in silent mode without any user interaction.

The deployment utility configuration file

You can pass parameters to the deployment utility using command-line options, or by using a configuration file that you pass as a parameter with the `-config` command-line option. You can also use both command-line options and a configuration file. Some examples on using **ifxdeploy** with command-line options and a configuration file on Linux and Windows are given below:

- **Example 1: Invoking the deployment utility with command-line options and a configuration file on Linux and UNIX:**

  ```
  ifxdeploy -y -verbose -config ifxdeploy.conf -file
   /opt/IBM/Informix/demo_on.tgz –start 180
  ```

- **Example 2: Invoking the deployment utility with command-line options and a configuration file on Windows:**

  ```
  ifxdeploy.exe -y -verbose -config ifxdeploy.conf -file
   C:\informixtemp\demo_on.zip –start 180
  ```

The following points summarize key information about the deployment utility configuration file:

- A sample deployment utility configuration file named `ifxdeploy.conf` is included on the Informix server at `$INFORMIXDIR/etc` on Linux and UNIX and at `%INFORMIXDIR%\etc` on Windows.
- You can also specify instance-specific information for the instance that is getting deployed in the configuration file:

  INFORMIXSERVER sets the primary server name.

  SERVERNUM sets the primary server port number.

  INFORMIXSQLHOSTS sets the value for the INFORMIXSQLHOSTS environment variable.

  ONCONFIG sets the `onconfig` file. If not specified and the **`ONCONFIG`** environment variable is not set, a new `onconfig` file is created based on the `onconfig.std` file template.
- Use BEGIN ALIAS and END ALIAS statements to define additional server names and listeners. Each alias results in a new SQLHOSTS entry and a new value for the DBSERVERALIASES configuration parameter in the `onconfig` file.

- Use the BEGIN ONCONFIG and END ONCONFIG statements to add to or override configuration parameters values specified in the onconfig file.
- **Example 3: Sample deployment utility configuration file entries for Windows**

```
INFORMIXSERVER  demo_on
PROTOCOL1 onsoctcp
PORT1 9088
SERVERNUM 1

BEGIN ALIAS
SERVERNAME alias1
PROTOCOL drsoctcp
PORT 9091
END ALIAS

INFORMIXDIR "C:\Program Files\IBM\Informix"

RELOCATE C:\IFMXDATA\demo_on=C:\IFMXDATANEW\demo_on

ROOTPATH C:\IFMXDATANEW\demo_on\rootdbs_dat.000

BEGIN ONCONFIG
BUFFERPOOL default,buffers=1000,lrus=8,lru_min_dirty=50.000000,
lru_max_dirty=60.500000
MAX_PDQPRIORITY 80
END ONCONFIG
```

Dynamic chunk relocation

The deployment utility provides the option to relocate the dbspaces to a new location of your choice. This option is particularly useful when the drives or the directory structures of the source template computer and the target computer are different. An example of this would be if the data spaces are on/data1 on the template computer and you want to relocate them to /data2 on the target computer.

You can set the new location of the dbspaces either by using the -relocate command-line parameter with **ifxdeploy**, or by using the configuration parameter RELOCATE in the ifxdeploy.conf file. You can specify a single new parent directory or map multiple separate chunk paths to different locations.

Example 4: Command-line argument to relocate all chunk paths to a single directory /data2/demo_on

```
-relocate /data2/demo_on
```

You can set the new location of the dbspaces either by using the -rootpath command-line parameter with **ifxdeploy**, or by using the configuration parameter ROOTPATH in the ifxdeploy.conf file. If you relocate dbspaces, you must also relocate the root dbspace.

Example 5: Command-line argument to specify a new rootpath

```
-rootpath /data2/demo_on/online_root
```

Automating deployments of reduced-footprint Informix on Linux

This section contains information about installing Informix, creating an instance, using the deployment assistant to create a snapshot of the installation, and silently deploying the snapshots on Linux.

Perform Informix installation and create an instance on Linux

The first task in the embeddability tutorial is to complete an installation and create an IBM Informix instance.

Install Informix on the template computer to prepare the master copy for later deployment. A custom installation is recommended because you can choose the features to install. This is one good way to reduce the footprint of Informix. Some features are mutually dependent and must be installed with one another. The good news is that the installation application manages these interdependencies.

Tip: You will be able to further reduce the footprint of the installation when you create a snapshot of the Informix instance with the deployment assistant by clearing features and packages that you do not want to include in the snapshot.

The embeddability tutorial for Linux is based on the following installation scenario:

- Informix 11.70 is installed in the default directory `/opt/IBM/informix`
- An instance with the name `demo_on` is created

Important: The `/opt/IBM/informix` installation path and an instance named `demo_on` are assumptions made for the tutorial. You can install Informix in a directory of your choice and create an instance with the name of your choice.

Creating a snapshot for deployment on Linux

The second task in the embeddability tutorial is to create a snapshot of the IBM Informix install binary files and the dbspaces on the template computer by using the deployment assistant.

Prerequisites:

- An Informix installation and an instance of it are on a template computer.
- You must be logged in to the template computer as user **informix**.
- Save copies of `$INFORMIXDIR/bin/ifxdeploy` and `$INFORMIXDIR/etc/ifxdeploy.conf` outside of the $INFORMIXDIR directory. (You will need to have these copies outside of $INFORMIXDIR when you deploy the snapshot on the target computer.)
- The Informix instance from which you want to create the snapshot is running.

1. Start a terminal window.
2. Set the **INFORMIXDIR** and **INFORMIXSERVER** environment variables.

   For example:

   ```
   export INFORMIXDIR=/opt/IBM/informix
   export INFORMIXSERVER=demo_on
   ```
3. Start the deployment assistant with the following commands:

   ```
   cd $INFORMIXDIR/bin
   ifxdeployassist
   ```
4. Enter database server connection parameters in the first deployment assistant GUI window. If you are completing this task as part of the embeddability tutorial, specify the following information:
   a. Instance Name: `demo_on`
   b. Host Name: `localhost`
   c. Port Number: `9088`
   d. Connection Type: `SQLI`

5. Specify the full path name of the snapshot file. Enter `/home/informix/Desktop/demo_on` if you are completing this task as part of the embeddability tutorial.
6. Optional: Clear the features or packages that you do not want to include in the snapshot to reduce the footprint.

   Important: The deployment assistant does not enforce any interdependencies between components of a functional instance.
7. If you are prompted with the Data Spaces window, decide whether to include the dbspaces associated with the source Informix instance in the snapshot. Keep the **Include the Data Spaces** box checked if you are doing this task as part of the embeddability tutorial.

   Tip: Record the full path names of the dbspaces if you select to include them in the snapshot. You will need to know the locations of the dbspaces when you deploy the snapshot.
8. Review the deployment configuration summary.
9. Record the information displayed in the Deployment Summary window and click **Close** . Save the information that you have recorded because you will need to know the snapshot locations for the next step.
10. Save the snapshots that you have created with the deployment assistant, the `$INFORMIXDIR/bin/ifxdeploy` file, and the `$INFORMIXDIR/etc/ifxdeploy.conf` file on external media (for example, a flash drive or CD) for future deployment. For the embeddability tutorial: save the database server snapshot as `demo_on.tar.gz` and the dbspaces snapshot as `demo_on_db.tar.gz`.

Silently deploy snapshots on Linux

The third task of the embeddability tutorial is to silently deploy an IBM Informix snapshot.

Deploying an Informix instance and its dbspaces is a two-step process:

1. Deploy dbspaces from the snapshot that was created using the deployment assistant.
2. Deploy the server instance, optionally relocate the dbspaces, and start the instance by using the deployment utility.

You can automate this process by using a simple shell script. The **ifx_silent_deploy.sh** script is an example shell script which serves this purpose.

Important: Refer to the shell script posted on the Technote at http://www.ibm.com/support/docview.wss?uid=swg21446737. The script contains comments that provide information about the purpose of each step. You can edit and customize the script to fit any deployment needs.

Prerequisites:

- You must be logged in to the target computer as root user.
- You must have a snapshot of an Informix instance and any associated dbspaces that was created with the deployment assistant.
- The **INFORMIXDIR** environment variable must be set.
- A folder named `informixtemp` must exist on the target computer.
- All the dbspaces' chunks must be cooked files, and they must be located in a single directory.

- The following files and utilities must be in the `informixtemp` folder on the target computer:

  Informix instance snapshot (for example: **demo_on.tgz**)

  dbspaces snapshot (for example: **demo_on_db.tgz**)

  deployment utility (**ifxdeploy**)

  deployment utility configuration file (`ifxdeploy.conf`)

  the **ifx_silent_deploy.sh** shell script to complete silent deployment

1. Run the **chmod +x ifx_silent_deploy.sh** command to give execute permission to the root user to run the **ifx_silent_deploy.sh** script.
2. Configure the following `ifxdeploy.conf` file parameters. (See "Sample ifxdeploy.conf file for Linux" on page 3-7 for an example to use for the tutorial.)
   a. Set the INFORMIXSERVER, ONCONFIG, and INFORMIXSQLHOSTS parameters. Alternatively, these parameters can be set as environment variables.
   b. Set the user **informix** password on the target computer in the INFORMIXPASSWORD parameter of the `ifxdeploy.conf` file.
   c. Optional: Set other parameters in the `ifxdeploy.conf` file as needed for your environment.
3. Use the following information about the **ifx_silent_deploy.sh** script to silently deploy the Informix instance. After the syntax information, there are two examples given that indicate specific values if you are completing the embeddability tutorial.

   Syntax:

   ```
   ifx_silent_deploy.sh <relocate_option> <srvpkg> <dbspkg> <currloc> [<newloc>]
   ```

   - `relocate_option`: use `relocate` to relocate dbspaces and `norelocate` to not relocate dbspaces
   - `srvpkg`: name of the server package
   - `dbspkg`: name of the dbspaces package
   - `currloc`: current location of the dbspaces
   - `newloc`: new deployment location of the dbspaces when relocating. This argument is not required if you are not relocating dbspaces.

   Important: The following commands are only examples, such as if you are using this documentation with sample values to complete the embeddability tutorial. The names of the snapshot files and location of dbspaces might be different in your instance.

   - *To silently deploy an Informix instance and relocate the dbspaces:*

     ```
     ifx_silent_deploy.sh relocate demo_on.tgz demo_on_db.tgz
      /data/IBM/informix/demo/server  /opt/IBM/data/demo_on
     ```

     This command deploys the base server of the snapshot, relocates the dbspaces from `/data/IBM/informix/demo/server` to `/opt/IBM/data/demo_on`, and starts the deployed instance.

   - *To silently deploy an Informix instance without relocating the dbspaces:*

     ```
     ifx_silent_deploy.sh norelocate demo_on.tgz demo_on_db.tgz
      /data/IBM/informix/demo/server
     ```

     This command deploys the Informix base server instance, deploys associated dbspaces to `/data/IBM/informix/demo/server`, and starts the instance.

See the `informixtemp/ifxdeploy.log` file for messages that are logged by the deployment utility. If the Informix instance does not start automatically, the

probable cause is that during creation of the snapshot you removed a feature or package that is required by the base server instance to run.

The deployed instance does not require the `ifxdeploy.conf` file to function. You can delete the `ifxdeploy.conf` file from the `informixtemp` directory after deploying the Informix instance.

Sample ifxdeploy.conf file for Linux

This topic contains an example of an `ifxdeploy.conf file` that has been set for configuring an instance snapshot on Linux.

```
##############################################################################
#  Licensed Material - Property Of IBM
#
#  "Restricted Materials of IBM"
#
#  IBM Informix
#  Copyright IBM Corporation 2009-2010 All rights reserved.
#
#  Title: ifxdeploy.conf
#  Description: Configuration file for the IDS Deployment Utility
#
#  Uncomment any values that you want to change from the default values.
#
#  Note that any parameters set on the command line will override these values.

# Primary server values
#     - These values define the primary server name, protocol, and port.
#     - Use the BEGIN ALIAS section to define additional sever names and
#       protocols (such as DRDA).

# INFORMIXSERVER - Set the primary server name, or set it as an environment
#                  variable or command line parameter.
INFORMIXSERVER demo_on

# PROTOCOL1 - Set the primary protocol (the sqlhosts NETTYPE field) for the
#             primary server.
#           - Values: onsoctcp, onipcnmp
#PROTOCOL1 onsoctcp

# SQLIPORT - Set the SQLI listening port for the primary server (not needed for
#            onipcnmp).
#          - Range: 1-65535
#SQLIPORT 9088

# DRDAPORT - Set the primary DRDA listening port for IBM Data Server Driver
#            communication.
#          - Range: 1-65535
#DRDAPORT 9089

# SERVERNUM - Set the primary server number (the value for the SERVERNUM
#             configuration parameter).
#           - Range: 0-255
SERVERNUM 0

# INFORMIXSQLHOSTS - Set a value for the INFORMIXSQLHOSTS environment
#                    variable. On UNIX this value specifies the sqlhosts file
#                    (default is $INFORMIXDIR/etc/sqlhosts). On Windows, this
#                    value is generally not used but can be used to point to a
#                    remote machine (for example, \\machinename) whose registry
#                    contains SQLHOSTS information.
INFORMIXSQLHOSTS sqlhosts.demo_on

# Define additional server names and listeners with the BEGIN/END ALIAS
# statements.
```

```
# Each ALIAS results in a new SQLHOSTS entry and a new value for the
# DBSERVERALIASES configuration parameter in the onconfig file.
# For example:
#BEGIN ALIAS
#SERVERNAME alias1
#PROTOCOL drsoctcp
#PORT 9091
#OPTIONS # optional SQLHOSTS parameters (for example, b=32767 to set buffers)
#END ALIAS

# INFORMIXDIR - Set the location of the installation directory.
#             - Alternatively, set the INFORMIXDIR environment variable.
INFORMIXDIR /opt/IBM/informix

# ONCONFIG - Set the onconfig file.
#          - If not specified and the ONCONFIG environment variable is not set,
#            a new onconfig file is created based on the onconfig.std file.
ONCONFIG onconfig.demo_on

# SNAPSHOT - Set the location of the compressed archive. This parameter is the
#            equivalent to the -file command line option. The archive must be a
#            .tgz file on UNIX or Linux and a .zip file on Windows. Only set
#            this value if you are supplying a compressed snapshot of an IDS
#            instance.
#SNAPSHOT

# RELOCATE  - Set to the new location of dbspace chunks.
#          Use one of these methods or a combination of methods 2 and 3:
#          - Method 1: new_path (relocates all chunks to the specified path)
#          - Method 2: old_path=new_path (relocates only chunks
#                                created in the old path to the new path)
#          - Method 3: old_path,old_offset=new_path,new_offset;
#                                (relocates chunks and moves offsets)
#          You can specify multiple paths with methods 2 and 3 by
#          separating old and new path sets with a semicolon (;).
#
#RELOCATE

# Authentication values
# INFORMIXPASSWORD - Set the password for the informix user.
#                  -  If not set, can be supplied on command line or
#                     interactively.
#                  - Not required if the SYSTEM parameter is set to 2.
INFORMIXPASSWORD inform1x

# SYSTEM - Windows only - Set the IDS service to log on as the Windows
#          Local System user.
#        - Values:
#            0 - IDS service logs on as the informix user.
#            1 - IDS service logs on as the Local System user but creates the
#                informix user.
#            2 - Do not create the informix user.
#SYSTEM 0

# Logging parameters
# LOGFILE - Set the file for Deployment  Utility errors and messages.
LOGFILE /usr/software/informixtemp/ifxdeploy.log

# LOGLEVEL - Set the amount of information to write to the log.
#
# 1  - FATAL - only print fatal errors.
# 3  - WARNING - print warnings and fatal errors.
# 5  - INFO - print informational messages, warnings, and fatal errors.
# 10 - DEBUG - print debugging information and all other messages.
#
LOGLEVEL 5
```

```
# SILENT - Set to 1 to prevent console output sot that errors and
#                    messages only appear in the log file.
#        - Range: 0,1
#SILENT 1

# FORCE - Set to 1 to overwrite existing settings
#       - Range: 0,1
#FORCE 0

# INSTALLDRIVE - Windows only - Set to the drive where data spaces will be
#                created
#              - Range: C-Z
#INSTALLDRIVE C

# ROOTPATH   - Set to the path for the root dbspace
#            - Default is \ifmxdata\$INFORMIXSERVER\rootdbs_dat.000
#ROOTPATH

# WIN6432 - Windows only - Set this to 1 if installing a 32-bit version of IDS
#           on a 64-bit Windows operating system.
#           - Range: 0,1
#WIN6432 0

# Onconfig customization
# Use the BEGIN ONCONFIG and END ONCONFIG statements to add or override
# configuration parameters values in the onconfig file.
# Use instead of providing an onconfig file.
# Example:
#BEGIN ONCONFIG
#LOCKS 10000
#END ONCONFIG
BEGIN ONCONFIG
BUFFERPOOL default,buffers=1000,lrus=8,lru_min_dirty=50.000000,
  lru_max_dirty=60.500000
BUFFERPOOL size=2K,buffers=1000,lrus=8,lru_min_dirty=50.000000,
  lru_max_dirty=60.000000
MAX_PDQPRIORITY 80
END ONCONFIG


#CLONE         - Deploy a clone of a source server. The information for source
#                server is specified within BEGIN CLONE / END CLONE statements
#              - Range: 0 (no clone deployment), 1 (clone deployment)
#              - Default: 0
#CLONE 0

#Define values for cloning a server within BEGIN CLONE / END CLONE statements
#BEGIN CLONE

# Source server information
# SOURCESERVER - Set to the name of source server
#SOURCESERVER
#SOURCEIPADDR  - Set to the IP address of source host
#SOURCEIPADDR
#SOURCEPORT    - Set to the listening port of source server
#SOURCEPORT

# Clone server infomation
#CLONEIPADDR   - Set to the IP address of clone host
#CLONEIPADDR
#DISPOSITION   - Set the final disposition of clone server
#              - Values: HDR, RSS or ER
#              - Default: Standard
#DISPOSITION
#TARGETSIZE    - Set the size of clone server
#              - Values: tiny, small, medium or large
#              - Default: Same as source server
```

```
#TARGETSIZE
#USELOCAL      - Use local configuration after merging with source configuration
#              - Range: 0 (use source configuration), 1(use local configuration)
#              - Default: 0
#USELOCAL
#TRUSTED       - User is trusted on source server
#              - Range: 0 (user is trusted), 1 (user is not trusted)
#              - Default: 0
#TRUSTED
#USERNAME      - Set to the name of user for connecting to source server
#              - Required if user is not trusted
#USERNAME
#PASSWORD      - Set to the password for above user name
#              - Required if user is not trusted
#PASSWORD

#END CLONE
```

Automating deployments of reduced-footprint Informix on Windows

This section contains information about installing Informix, creating an instance, using the deployment assistant to create a snapshot of the installation, and silently deploying the snapshots on Windows.

Perform Informix installation and create an instance on Windows

The first task in the embeddability tutorial is to complete an installation and create an IBM Informix instance.

Install Informix on the template computer to prepare the master copy for later deployment. A custom installation is recommended because you can choose which features to install. This is one good way to reduce the footprint of Informix. Some features are mutually dependent and must be installed with one another. The good news is that the installation application manages these interdependencies.

Tip: You will be able to further reduce the footprint of the installation when you create a snapshot of the Informix instance with the deployment assistant by clearing features and packages that you do not want to include in the snapshot.

The embeddability tutorial for Windows is based on the following installation scenario:

- Informix 11.70 is installed in the default directory `C:\Program Files\IBM\Informix`
- An instance with the name `demo_on` is created

Important: The `C:\Program Files\IBM\Informix` installation path and an instance named `demo_on` are assumptions made for the tutorial. You can install Informix in a directory of your choice and create an instance with the name of your choice.

Creating a snapshot for deployment on Windows

The second task in the embeddability tutorial is to create a snapshot of the IBM Informix install binary files and the dbspaces on the template computer by using the deployment assistant.

Prerequisites:

- An Informix installation and an instance of it are on a template computer.

- You must be logged in to the template computer as user **informix** or as a user with Administrator privileges.
- In order to package dbspaces, the user must have CONNECT privileges to the **sysadmin** database. By default, only the user **informix** has this permission. If you log in as a user other than user **informix**, you must grant the CONNECT privilege to the **sysdamin** database for this user. This is a security mechanism implemented to protect the data.
- Save copies of the files `%INFORMIXDIR%\bin\ifxdeploy.exe` and `%INFORMIXDIR%\etc\ifxdeploy.conf` files in a folder outside `%INFORMIXDIR%`. You will need these files when you deploy Informix on the target computer.
- The Informix instance from which you want to create the snapshot is running.

1. Start a command window.
2. Set the **INFORMIXDIR** and **INFORMIXSERVER** environment variables.

   For example, for the embeddability tutorial, specify:

   ```
   set INFORMIXDIR=C:\Program Files\IBM\Informix
   set INFORMIXSERVER=demo_on
   ```
3. Start the deployment assistant with the following command.

   For example, for the embeddability tutorial, specify:

   ```
   cd C:\Program Files\IBM\Informix\bin
   ifxdeployassist
   ```
4. Verify that the deployment assistant detects the Informix instance to be used as a template for the snapshot.
5. Specify the full path name of the snapshot to be created. For the embeddability tutorial, specify `C:\Documents and Settings\Administrator\Desktop\demo_on.zip`.
6. Optional: Clear the features or packages that you do not want to include in the snapshot to reduce the footprint.

   Important: The deployment assistant does not enforce any interdependencies between components of a functional instance.
7. If you are prompted with the Data Spaces window, decide whether to include the dbspaces associated with the template Informix instance in the snapshot. Keep the **Include the Data Spaces** box checked if you are creating a snapshot following the embeddability tutorial.

   Tip: Record the full path names of the dbspaces if you select to include them in the snapshot. You will need to know the locations of the dbspaces when you deploy the snapshot.
8. Review the deployment configuration summary.
9. Record the information displayed in the Deployment Summary window and click **Close** . Save the information that you have recorded because you will need to know the snapshot locations for the next step.
10. Save the snapshots that you created with the deployment assistant, the `%INFORMIXDIR%\bin\ifxdeploy.exe` file and the `%INFORMIXDIR%\etc\ifxdeploy.conf` file on external media (for example, a flash drive or CD). You will need these for the next task.

    For the embeddability tutorial: save the database server snapshot as **demo_on.zip** and the dbspaces snapshot as **demo_on_db.zip**.

Silently deploy Informix on Windows

The third task of the embeddability tutorial is to silently deploy an IBM Informix snapshot.

Deploying an Informix instance and its dbspaces is a two-step process:

1. Deploy dbspaces from the snapshot that was created using the deployment assistant.
2. Deploy the server instance, optionally relocate the dbspaces, and start the instance by using the deployment utility.

You can automate this process by using a simple batch script. The **ifx_silent_deploy.cmd** script is an example batch script which serves this purpose.

Important: Refer to the batch script posted on the Technote at http://www.ibm.com/support/docview.wss?uid=swg21446737. The script contains comments that provide information about the purpose of each step. You can edit and customize the script to fit any deployment needs.

Prerequisites:

- You must be logged in to the target computer as an Administrator user.
- You must have a snapshot of an Informix instance and any associated dbspaces that was created with the deployment assistant.
- Installation of 7-zip on the target computer. If you need to install 7-zip, download and install it on the target computer at the default location `C:\Program Files\7-Zip`.
- A folder named `informixtemp` must exist on the `C:\` drive of the target computer.
- The **INFORMIXDIR** environment variable must be set.
- All the dbspaces' chunks must be cooked files, and they must be located in a single directory.
- The following files and utilities must be in the `C:\informixtemp` folder on the target computer:

  Informix server instance snapshot (Example: `demo_on.zip`)

  dbspaces snapshot (Example: `demo_on_db.zip`)

  deployment utility (`ifxdeploy.exe`)

  deployment utility configuration file (`ifxdeploy.conf`)

  the `ifx_silent_deploy.cmd` script to complete silent deployment

1. Configure the following `ifxdeploy.conf` file parameters. See "Sample ifxdeploy.conf file for Windows" on page 3-13 for an example to use for the tutorial.
   a. Set the INFORMIXSERVER and ONCONFIG parameters. Alternatively, these parameters can be set as environment variables.
   b. Set the user **informix** password on the target computer in the INFORMIXPASSWORD parameter of the `ifxdeploy.conf` file.
   c. Optional: Set other parameters in the `ifxdeploy.conf` file as needed for your environment.
2. Use the following information about the **ifx_silent_deploy.cmd** script to silently deploy the Informix instance. After the syntax information, there are two examples given that indicate specific values if you are completing the embeddability tutorial.

   Syntax:

```
ifx_silent_deploy.cmd <relocate_option> <srvpkg> <dbspkg> <currloc> [<newloc>]
```

- relocate_option: use relocate to relocate dbspaces and norelocate to not relocate dbspaces
- srvpkg: name of the server package (for example: **demo_on.zip**)
- dbspkg: name of the dbspaces package (for example: **demo_on_db.zip**)
- currloc: current location of the dbspaces
- newloc: new deployment location of the dbspaces when relocating. This argument is not required if you are not relocating dbspaces.

Important: The following commands are only examples, such as if you are using this documentation with sample values to complete the embeddability tutorial. The names of the snapshot files and location of dbspaces might be different in your instance.

- *To silently deploy an Informix instance and relocate the dbspaces:*

  ```
  ifx_silent_deploy.cmd relocate demo_on.zip demo_on_db.zip
    C:\IFMXDATA\demo_on C:\IFMXDATANEW\demo_on
  ```

  This command deploys the Informix instance, relocates dbspaces from C:\IFMXDATA\demo_on to C:\IFMXDATANEW\demo_on and starts the instance.
- *To silently deploy an Informix instance without relocating the dbspaces:*

  ```
  ifx_silent_deploy.cmd norelocate demo_on.zip demo_on_db.zip
    C:\IFMXDATA\demo_on
  ```

  This command deploys the Informix instance, deploys dbspaces to C:\IFMXDATA\demo_on, and starts the instance.

See the C:\informixtemp\ifxdeploy.log file for messages that are logged by the deployment utility. If the Informix instance does not start automatically, the probable cause is that during creation of the snapshot you removed a feature or package that is required by the base server instance to run.

You can delete the ifxdeploy.conf file from C:\informixtemp after deploying the Informix instance.

Sample ifxdeploy.conf file for Windows

This topic contains an example of an ifxdeploy.conf file that has been set for configuring an instance snapshot on Windows.

```
#############################################################################
#  Licensed Material - Property Of IBM
#
#  "Restricted Materials of IBM"
#
#  IBM Informix
#  Copyright IBM Corporation 2009 All rights reserved.
#
#  Title: ifxdeploy.conf
#  Description: Configuration file for the IDS Deployment Utility
#
#  Uncomment any values that you want to change from the default values.
#
#  Note that any parameters set on the command line will override these values.

# Primary server values
#     - These values define the primary server name, protocol, and port.
#     - Use the BEGIN ALIAS section to define additional sever names and
#       protocols (such as DRDA).

# INFORMIXSERVER - Set the primary server name, or set it as an environment
#                  variable or command line parameter.
```

```
INFORMIXSERVER  demo_on

# PROTOCOL1 - Set the primary protocol (the sqlhosts NETTYPE field) for the
#             primary server.
#           - Values: onsoctcp, onipcnmp
#PROTOCOL1 onsoctcp

# PORT1 - Set the primary listening port for the primary server (not needed for
#         onipcnmp).
#           - Range: 1-32767
#PORT1 9088

# SERVERNUM - Set the primary server number (the value for the SERVERNUM
#             configuration parameter).
#           - Range: 0-255
SERVERNUM 0


# INFORMIXSQLHOSTS - Set a value for the INFORMIXSQLHOSTS environment
#                    variable. On UNIX this value specifies the sqlhosts file
#                    (default is $INFORMIXDIR/etc/sqlhosts). On Windows, this
#                    value is generally not used but can be used to point to a
#                    remote machine (for example, \\machinename) whose registry
#                    contains SQLHOSTS information.
#INFORMIXSQLHOSTS

# Define additional server names and listeners with the BEGIN/END ALIAS
# statements.
# Each ALIAS results in a new SQLHOSTS entry and a new value for the
# DBSERVERALIASES configuration parameter in the onconfig file.
# For example:
#BEGIN ALIAS
#SERVERNAME alias1
#PROTOCOL drsoctcp
#PORT 9091
#OPTIONS # optional SQLHOSTS parameters (for example, b=32767 to set buffers)
#END ALIAS

# INFORMIXDIR - Set the location of the installation directory.
#             - Alternatively, set the INFORMIXDIR environment variable.
INFORMIXDIR "C:\Program Files\IBM\Informix"

# ONCONFIG - Set the onconfig file.
#          - If not specified and the ONCONFIG environment variable is not set,
#            a new onconfig file is created based on the onconfig.std file.
ONCONFIG ONCONFIG.demo_on

# SNAPSHOT - Set the location of the compressed archive. This parameter is the
#            equivalent to the -file command line option. The archive must be a
#            .tgz file on UNIX or Linux and a .zip file on Windows. Only set
#            this value if you are supplying a compressed snapshot of an IDS
#            instance.
#SNAPSHOT c:\informixtemp\demo_on.zip

# RELOCATE  - Set to the new location of dbspace chunks.
#          Use one of these methods or a combination of methods 2 and 3:
#          - Method 1: new_path (relocates all chunks to the specified path)
#          - Method 2: old_path=new_path (relocates only chunks
#                                created in the old path to the new path)
#          - Method 3: old_path,old_offset=new_path,new_offset;
#                                (relocates chunks and moves offsets)
#          You can specify multiple paths with methods 2 and 3 by
#          separating old and new path sets with a semicolon (;).
#
#RELOCATE
#RELOCATE C:\IFMXDATANEW\demo_on
```

```
# Authentication values
# INFORMIXPASSWORD - Set the password for the informix user.
#                  -  If not set, can be supplied on command line or
#                     interactively.
#                  - Not required if the SYSTEM parameter is set to 2.
INFORMIXPASSWORD inform1x

# SYSTEM - Windows only - Set the IDS service to log on as the Windows
#          Local System user.
#        - Values:
#             0 - IDS service logs on as the informix user.
#             1 - IDS service logs on as the Local System user but creates the
#                 informix user.
#             2 - Do not create the informix user.
#SYSTEM 0

# Logging parameters
# LOGFILE - Set the file for Deployment  Utility errors and messages.
#LOGFILE
LOGFILE C:\informixtemp\ifxdeploy.log

# LOGLEVEL - Set the amount of information to write to the log.
#
# 1  - FATAL - only print fatal errors.
# 3  - WARNING - print warnings and fatal errors.
# 5  - INFO - print informational messages, warnings, and fatal errors.
# 10 - DEBUG - print debugging information and all other messages.
#
LOGLEVEL 5

# SILENT - Set to 1 to prevent console output sot that errors and
#                    messages only appear in the log file.
#        - Range: 0,1
#SILENT 1

# FORCE - Set to 1 to overwrite existing settings
#       - Range: 0,1
#FORCE 0

# INSTALLDRIVE - Windows only - Set to the drive where data spaces will be
#                created
#              - Range: C-Z
#INSTALLDRIVE C

# ROOTPATH   - Set to the path for the root dbspace
#            - Default is \ifmxdata\$INFORMIXSERVER\rootdbs_dat.000
#ROOTPATH
#ROOTPATH C:\IFMXDATANEW\demo_on\rootdbs_dat.000

# WIN6432 - Windows only - Set this to 1 if installing a 32-bit version of IDS
#           on a 64-bit Windows operating system.
#           - Range: 0,1
#WIN6432 0

# Onconfig customization
# Use the BEGIN ONCONFIG and END ONCONFIG statements to add or override
# configuration parameters values in the onconfig file.
# Use instead of providing an onconfig file.
BEGIN ONCONFIG
BUFFERPOOL default,buffers=1000,lrus=8,lru_min_dirty=50.000000,
  lru_max_dirty=60.000000
BUFFERPOOL size=4K,buffers=1000,lrus=8,lru_min_dirty=50.000000,
  lru_max_dirty=60.000000
MAX_PDQPRIORITY 50
END ONCONFIG
```

Chapter 4. Manage memory in embedded environments

You can manage memory in embedded environments by enabling the database server to automatically rotate and delete online message log files, configuring the actions that the server takes to continue processing when memory is critically low, and enabling the server to reserve a specific amount of memory for use when critical rollback activities are needed and the server has limited free memory.

Optimize storage

You can optimize storage by compressing data, configuring the server to add more storage space automatically, and defragmenting partitions. Compressing data reduces the amount of disk space needed for your applications.

Use SQL administration API commands to:

- Compress data in table or fragment rows
- Consolidate the free space that remains in the table or fragment
- Return the free space to the dbspace.
- Implement automatic space management
- Defragment partitions to non-contiguous extents

Manage message logs in an embedded environment

You can enable the database server to automatically rotate and delete online message log files, and you can specify the maximum number of online, ON-Bar activity, or ON-Bar debug message log files to rotate. The Scheduler tasks that rotate and delete these message log files are useful for embedded applications, because they reduce DBA or system administrator requirements for managing the logs.

Additionally, you can run SQL administration API commands to rotate, remove the contents of (*truncate*), or delete one of these message log files. For example, you might want to run a truncate or delete command when performing a load operation.

Enabling and disabling the tasks that automatically rotate message log files

Before the database server can run the built-in Scheduler tasks that automatically rotate message log files, you must enable each task that rotates a message log file. You only need to enable each task once. You can also disable an enabled task.

To enable a task for rotating logs:

Run the following SQL statement, which updates the value of the **tk_enable** column of the **ph_task** table to `t` (true) for the message log file where the **tk_name** column is the name of the message log, as follows:

```
UPDATE ph_task SET tk_enable = t , tk_frequency = frequency
WHERE tk_name = name_of_message_log task";
```

The name of the message log must be bar_act_log_rotate, bar_debug_log_rotate, or online_log_rotate.
For example, to enable the **online_log_rotate** task and set the frequency of the task to every 30 days, specify:

```
DATABASE sysadmin;
UPDATE ph_task SET tk_enable = "t" ,
 tk_frequency = INTERVAL (30) DAY TO DAY
WHERE tk_name = "online_log_rotate";
```

You can also specify tk_frequency = tk_frequency if you do not want to accept the default frequency (3:00 A.M. every 30 days) when you enable the task.

The only possible values for tk_enable are t and f (true and false).

To disable the running of a message log rotation task, set tk_enable to f, as shown in the following example:

```
UPDATE ph_task SET tk_enable = "f" ,
 tk_frequency = INTERVAL (30) DAY TO DAY
WHERE tk_name = "bar_debug_log_rotate";
```

You can also specify tk_frequency = tk_frequency if you want to accept the default frequency (3:00 A.M. every 30 days).

Tasks that automatically rotate message log files

A Scheduler task automatically rotates each of the common message log files (online.log, bar_act_log, and bar_debug_log). When the server rotates a log file, the server switches to a new log file and increments the ID numbers for the previous log files by one. When the maximum number of log files is reached, the log file with the highest ID is deleted.

The following table shows the tasks that automatically rotate message log files and the default frequency for when the tasks run:

Table 4-1. Message log files, tasks that rotate the files, and default frequency

Message log file	Task that rotates the file	Default frequency for when task runs
online.log	**online_log_rotate** task	3:00 A.M. every 30 days You can change the frequency, which is specified in the MAX_MSGPATH_VERSIONS parameter of the **ph_threshold** table.
bar_act_log	**bar_act_log_rotate** task	3:00 A.M. every 30 days You can change frequency, which is specified in the MAX_BAR_ACT_LOG_VERSIONS parameter of the **ph_threshold** table.
bar_debug_log	**bar_debug_log_rotate** task	3:00 A.M. every 30 days You can change frequency, which is specified in the MAX_BAR_DEBUG_LOG_VERSIONS parameter of the **ph_threshold** table.

You can change the maximum number of online, ON-Bar activity, or ON-Bar debug message log files that the database server retains. This information is stored in the MAX_MSGPATH_VERSIONS, MAX_BAR_ACT_LOG_VERSIONS, and MAX_BAR_DEBUG_LOG_VERSIONS parameters of the ph_threshold table.

Before the server can run the tasks that automatically rotate message log files, you must enable each task. You only need to enable each task once.

Configuring the frequency of automatic rotation of message log files

You can configure the frequency of the built-in Scheduler tasks that automatically rotate online message log files.

The default frequency of the **bar_act_log_rotate**, **bar_debug_log_rotate**, and **online_log_rotate** tasks is 3 A.M. every 30 days with a maximum of 12 log files, but you can configure the task to run more or less frequently.

Prerequisite:

If you did not enable the task for rotating the message log that you want to configure, do that now.

To configure a task for rotating logs to run more or less frequently:

Run the following SQL statements, where *days* is the number of days between each run:

```
DATABASE sysadmin;
UPDATE ph_task set tk_frequency = frequency
 WHERE tk_name = "task_name";
```

The task name must be `bar_act_log_rotate`, `bar_debug_log_rotate`, or `online_log_rotate`.
For example, to change the frequency of the **online_log_rotate** task specify:

```
DATABASE sysadmin;
UPDATE ph_task SET tk_frequency = INTERVAL (10)
DAY TO DAY WHERE tk_name = "online_log_rotate";
```

You can also disable the running of message lot rotation tasks, by setting `tk_enable` to `f` (false), as follows:

```
UPDATE ph_task SET tk_enable = f , tk_frequency = tk_frequency
WHERE tk_name = "online_log_rotate";
```

Configuring the maximum number of message log files to retain

You can change the maximum number of online, ON-Bar activity, or ON-Bar debug message log files that the database server retains. This information is stored in the MAX_MSGPATH_VERSIONS, MAX_BAR_ACT_LOG_VERSIONS, and MAX_BAR_DEBUG_LOG_VERSIONS parameters of the **ph_threshold** table.

By default, the database server retains 12 online, ON-Bar activity, and ON-Bar debug message log files.

To change the maximum number of online log files to retain:

Run the following SQL statements to specify the maximum value of log files to retain, where *threshold_name* is MAX_MSGPATH_VERSIONS (for the online log files), MAX_BAR_ACT_LOG_VERSIONS, or MAX_BAR_DEBUG_LOG_VERSIONS.

```
UPDATE ph_threshold
  SET value = "number"
  WHERE name = "threshold_name";
```

For example, to set the max number of online log files to keep to 20, specify:

```
UPDATE ph_threshold
  SET value = "20"
  WHERE name = "MAX_MSGPATH_VERSIONS";
```

Reducing the size of message log files

You can reduce the size of online, ON-Bar activity, or ON-Bar debug message log files by running SQL administration API commands that rotate, truncate, or delete the logs.

Manually rotating a message log file

To manually rotate a message log file, run the admin() or task() function with the **message log rotate** argument, the name of the message log, and the maximum message log version that the server will rotate, as follows:

```
EXECUTE FUNCTION task("message log rotate",
 "full_path_for_message_log_file",maximum_version);
```

For example, to rotate a maximum of 52 online.logfiles, specify:

```
execute function task("message log rotate",
 "/usr/informix/online.log",52);
```

Manually truncating a message log file

To manually truncate a message log file (remove the contents of the file) , run the admin() or task() function with the **message log truncate** argument and the full path name for the message log file, as follows:

```
EXECUTE FUNCTION task("message log truncate", "full_path_for_message_log_file");
```

For example, to remove the contents of the online.log file, specify:

```
EXECUTE FUNCTION task("message log truncate",
 "/usr/informix/online.log");
```

Manually deleting a message log file

To manually delete the entire message log file (remove the contents of the file and the file), run the admin() or task() function with the **message log delete** argument and the full path name for the message log file, as follows:

```
EXECUTE FUNCTION task("message log delete", "path_for_message_log_file");
```

For example, to delete the entire online.logfile, specify:

```
EXECUTE FUNCTION task("message log delete",
 "/usr/informix/online.log");
```

Reserve memory for critical activities in embedded environments

You can reserve a specific amount of memory for use when critical activities (such as rollback activities) are needed and the database server has limited free memory. When you reserve memory, critical activities, such as rollback activities, can complete even when you receive out-of-memory errors.

To reserve memory for critical activities, set the LOW_MEMORY_RESERVE configuration parameter to a specified value in kilobytes. You can also change the LOW_MEMORY_RESERVE value by using the **onmode -wf** or **onmode -wm** utility.

Maintain a targeted amount of memory in embedded applications

You can configure the database server to continue processing when memory is critically low, instead of returning an out of memory error.

You specify the criteria for terminating sessions based on idle time, memory usage, and other factors so that the targeted application can continue to process. Configuring the low memory response is useful for embedded applications that have memory limitations.

To configure a targeted amount of memory in embedded applications:

- Set the LOW_MEMORY_MGR configuration parameter to 1, which enables the automatic low memory management when the server starts.
- Set the threshold parameters for the amount of memory to maintain by using an SQL administration API command with the **scheduler lmm enable** argument.

To disable automatic low memory management, run an SQL administration API command with the **scheduler lmm disable** argument.

Appendix. Accessibility

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

Accessibility features for IBM Informix products

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

Accessibility features

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

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

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

Related accessibility information

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

IBM and accessibility

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.

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.
1623-14, Shimotsuruma, Yamato-shi
Kanagawa 242-8502 Japan

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

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

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

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

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

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

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

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

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

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

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

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

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

L

M

R

S

U

V

Printed in USA

SC27-3528-04