

Informix Product Family
Informix
Version 12.10

IBM Informix Embeddability Guide



Informix Product Family
Informix
Version 12.10

IBM Informix Embeddability Guide



Note

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

Edition

This edition replaces SC27-4519-00.

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

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

© Copyright IBM Corporation 2010, 2013.

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

Contents

| | |
|--|------------|
| Introduction | v |
| About this publication | v |
| Types of users | v |
| Example code conventions. | v |
| Additional documentation. | v |
| Compliance with industry standards | vi |
| Syntax diagrams | vi |
| How to read a command-line syntax diagram. | vii |
| Keywords and punctuation | viii |
| Identifiers and names. | viii |
| How to provide documentation feedback | ix |
| | |
| What's new for embeddability in IBM Informix, Version 12.10 | xi |
| | |
| Chapter 1. Overview of embedding Informix products | 1-1 |
| Customize the size of Informix | 1-1 |
| Customize a silent installation | 1-1 |
| Easy deployment | 1-2 |
| Synchronize connectivity information between servers and clients on Windows | 1-2 |
| Automated administration | 1-3 |
| Easy configuration in embedded environments | 1-3 |
| Specify the Informix environment in the onconfig file | 1-4 |
| Maintain Informix availability | 1-5 |
| | |
| Chapter 2. Deploying Informix software | 2-1 |
| The deployment utility | 2-1 |
| Planning for deploying Informix | 2-2 |
| Creating a snapshot with the deployment assistant | 2-2 |
| Creating a snapshot for deployment manually | 2-7 |
| Deploying a snapshot with the deployment utility | 2-7 |
| The ifxdeploy command: The deployment utility | 2-8 |
| The ifxdeploy.conf file: The deployment utility configuration file | 2-16 |
| The syncsqlhosts utility. | 2-22 |
| Removing a snapshot with the deployment utility | 2-23 |
| Deploying Informix and client products with RPM-based installation (Linux). | 2-24 |
| | |
| Chapter 3. Informix embeddability deployment tutorial | 3-1 |
| Introducing the Informix deployment assistant | 3-1 |
| Introducing the Informix deployment utility | 3-1 |
| Automating deployments of reduced-footprint Informix on Linux | 3-4 |
| Perform Informix installation and create an instance on Linux | 3-4 |
| Creating a snapshot for deployment on Linux | 3-4 |
| Silently deploy snapshots on Linux | 3-5 |
| Automating deployments of reduced-footprint Informix on Windows. | 3-7 |
| Perform Informix installation and create an instance on Windows | 3-7 |
| Creating a snapshot for deployment on Windows | 3-8 |
| Silently deploy Informix on Windows | 3-9 |
| | |
| Chapter 4. Manage memory in embedded environments | 4-1 |
| Optimize storage | 4-1 |
| Manage message logs in an embedded environment | 4-1 |
| Enabling and disabling the tasks that automatically rotate message log files | 4-1 |
| Tasks that automatically rotate message log files | 4-2 |
| Configuring the frequency of automatic rotation of message log files | 4-3 |

| | |
|---|------------|
| Configuring the maximum number of message log files to retain | 4-4 |
| Reducing the size of message log files | 4-4 |
| Reserve memory for critical activities in embedded environments | 4-5 |
| Maintain a targeted amount of memory in embedded applications. | 4-5 |
| Appendix. Accessibility | A-1 |
| Accessibility features for IBM Informix products | A-1 |
| Accessibility features. | A-1 |
| Keyboard navigation. | A-1 |
| Related accessibility information | A-1 |
| IBM and accessibility. | A-1 |
| Dotted decimal syntax diagrams | A-1 |
| Notices | B-1 |
| Privacy policy considerations | B-3 |
| Trademarks | B-3 |
| Index | X-1 |

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.

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 that is being discussed.

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

Additional documentation

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

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

Compliance with industry standards

IBM Informix products are compliant with various standards.

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

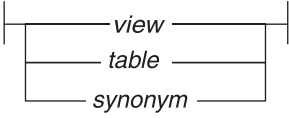
Syntax diagrams

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

Table 1. 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 1. Syntax Diagram Components (continued)

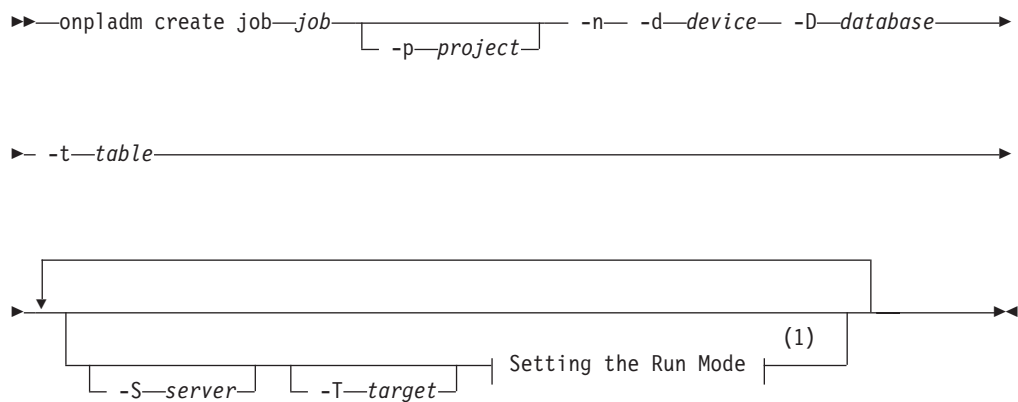
| Component represented in PDF | Component represented in HTML | Meaning |
|--|--|-----------------|
| <p>Table Reference</p>  | <p>Table Reference</p> <pre> --+-view-----+-- +-----table-----+ '-----synonym-----'</pre> | Syntax segment. |

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

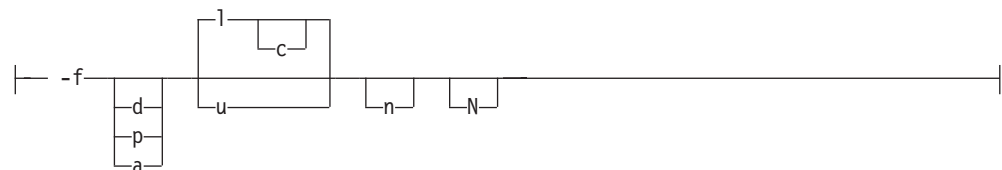


Notes:

1 See page Z-1

This diagram has a segment that is 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. Include **onpladm create job** and then the name of the job.
2. Optionally, include **-p** and then the name of the project.
3. Include 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 include 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 include **-f**, optionally include **d**, **p**, or **a**, and then optionally include **l** or **u**.
5. Follow the diagram to the terminator.

Keywords and punctuation

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

A keyword in a syntax diagram 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 other syntax diagrams. A variable in a syntax diagram, an example, or text, is shown in *lowercase italic*.

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

►►—SELECT—*column_name*—FROM—*table_name*—►►

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, complete 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.

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

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

For a complete list of what's new in this release, go to http://pic.dhe.ibm.com/infocenter/informix/v121/topic/com.ibm.po.doc/new_features_ce.htm.

Table 2. What's new for embeddability in Informix 12.10.xC1

| Overview | Reference |
|--|---|
| <p>Managing server connections on Windows operating systems</p> <p>On Windows operating systems, you now configure connectivity information for Informix servers by using the <code>sqlhosts</code> file, not the Windows registry. The file is installed in <code>%INFORMIXDIR%\etc\sqlhosts</code>. <code>%INFORMIXSERVER%</code>, and it uses the same format as the <code>sqlhosts</code> file on UNIX operating systems. The <code>sync_registry</code> Scheduler task automatically converts the connection information between the <code>sqlhosts</code> file format and the Windows registry format. The task runs every 15 minutes. You can manually convert the connection information between the <code>sqlhosts</code> file format and the Windows registry format by running the <code>syncsqlhosts</code> utility.</p> | <p>"The <code>syncsqlhosts</code> utility" on page 2-22</p> |
| <p>Set local environment variables for Informix instances</p> <p>You can set local environment variables in the <code>onconfig</code> file for an Informix instance. These settings are independent of the global or system environment variable settings on the computer. The settings can be used by the following utilities: oncheck, onclean, oninit, onload, onlog, onmode, onparams, onspaces, onstat, ontape, onunload. When you run the Informix utility you must specify the -FILE option before any other options.</p> <p>The -FILE option makes it easy to run Informix utilities on remote computers in embedded environments.</p> | <p>"Specify the Informix environment in the <code>onconfig</code> file" on page 1-4</p> |
| <p>Easily configure an embedded server</p> <p>You can now configure embedded servers so they require less setup:</p> <ul style="list-style-type: none">• Embed any environment variable into any configuration parameter value (easy porting to different servers)• Simplify configuration files (when Informix starts, it uses only a few critical parameters, and does not use the <code>onconfig</code> file)• Turn automatic tuning on or off with one new configuration parameter• Export and import configuration parameter values | <p>"Easy configuration in embedded environments" on page 1-3</p> |

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.

Related concepts:

 [Deployment wizard \(Installation Guide \(UNIX\)\)](#)
“The deployment utility” on page 2-1


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*.


Related concepts:

 [Non-root installation \(Installation Guide \(UNIX\)\)](#)

Related tasks:

 [Performing a silent installation of Informix and client products \(Installation Guide \(UNIX\)\)](#)

 [Performing an unattended Informix installation \(Mac OS X\) \(Installation Guide \(UNIX\)\)](#)

 [Performing a silent installation of Informix and client products \(Installation Guide \(Windows\)\)](#)

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.

Related tasks:

“Creating a snapshot with the deployment assistant” on page 2-2

“Creating a snapshot for deployment on Linux” on page 3-4

Synchronize connectivity information between servers and clients on Windows

On the Windows operating system, you must configure connectivity information for Informix Servers by using the `sqlhosts` file.

The `sqlhosts` file is installed in `%INFORMIXDIR%\etc\sqlhosts.%INFORMIXSERVER%`. The `sqlhosts` file on the Windows operating system uses the same format as the `sqlhosts` file on UNIX operating systems. The `sync_registry` task automatically converts the connection information between the `sqlhosts` file format and the Windows registry format.

When maintaining several client machines, you can update their registries by using the `syncsqlhosts` utility. For example, if a DRDA[®] connection is added to a server, all client machines can use this utility to update their registries as well.

Related reference:

“The syncsqlhosts utility” on page 2-22

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.


Related concepts:

 [The Scheduler \(Administrator's Guide\)](#)

Related reference:

 [Remote administration with the SQL administration API \(Administrator's Guide\)](#)

 [The ph_alert Table \(Administrator's Reference\)](#)

 [onconfig Portal: Configuration parameters by functional category \(Administrator's Reference\)](#)

Easy configuration in embedded environments

You can configure and tune the database server in embedded environments, regardless of the workload or information you have about resources.

Using one configuration parameter, the AUTO_TUNE configuration parameter, you can enable or disable all automatic tuning parameters. You can also replace multiple automatic tuning configuration parameters in your configuration file with the AUTO_TUNE configuration parameter.

You can export a snapshot of your current configuration parameters to a file. You can archive this file, use it as a configuration file, or import it to another running instance. Importing a configuration file is a convenient way of dynamically

modifying multiple tunable parameters at one time. If the file contains configuration parameters that cannot be tuned dynamically, the values of those parameters are ignored.

Related concepts:

- Configure the database server (Administrator's Guide)
- Configuration file used during initialization (Administrator's Guide)
“Specify the Informix environment in the onconfig file”

Related reference:

- `onmode -wf, -wm`: Dynamically change certain configuration parameters (Administrator's Reference)
- `onmode -we`: Export a file that contains current configuration parameters (Administrator's Reference)
- `onmode -wi`: Import a configuration parameter file (Administrator's Reference)
- `import config` argument: Import configuration parameter values (SQL administration API) (Administrator's Reference)
- `export config` argument: Export configuration parameter values (SQL administration API) (Administrator's Reference)
- `AUTO_TUNE` configuration parameter (Administrator's Reference)

Specify the Informix environment in the onconfig file

You can set local environment variables in the `onconfig` file for an Informix instance. These settings, which are independent of the global or system environment variable settings on the computer, can be used by the **oninit**, **oncheck**, **onclean**, **onload** and **onunload**, **onlog**, **onmode**, **onparams**, **onspaces**, **onstat**, and **ontape** utilities in embedded environments.


When you use the **-FILE** option with one of the specified utilities, you can:

- Start the database server from a remote machine without having to create specific environment files and wrappers for each instance or without having to set the environment in a shell.
- Define information for the server to use automatically instead of setting environment variables for an instance before you start the database server.
- Run an IBM Informix utility without having to set your environment explicitly in the shell before you run the utility.
- Have a single point to specify environment variables for instances in order to further embed the underlying implementation and installation of the database server.

Related concepts:

“Easy configuration in embedded environments” on page 1-3

Related tasks:

 [Setting local environment variables for utilities \(Administrator's Reference\)](#)

Related reference:

 [The -FILE option \(Administrator's Reference\)](#)

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.

Related concepts:

 [IBM Informix Enterprise Replication technical overview \(Enterprise Replication Guide\)](#)

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.

Related concepts:

“Introducing the Informix deployment utility” on page 3-1

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 deploy 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.

Related concepts:

“Customize the size of Informix” on page 1-1

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 server. 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.

Related concepts:

Chapter 3, “Informix embeddability deployment tutorial,” on page 3-1

Related tasks:

“Creating a snapshot with the deployment assistant”

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.

Related concepts:

“Easy deployment” on page 1-2

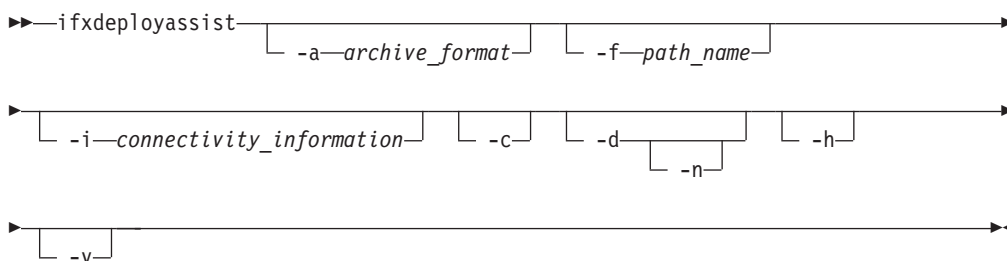
“Planning for deploying Informix” on page 2-2

“Introducing the Informix deployment assistant” on page 3-1

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 |
|--------------------------------|--|--|
| <code>-a archive_format</code> | <p>Specifies the archive format and compression type to save the snapshot.</p> <p>The following values can be used in the place of <i>archive_format</i>:</p> <ul style="list-style-type: none">• BZIP2 (tar archive with BZip2 compression)• GZIP (tar archive with GZIP compression)• TAR (tar archive with no compression)• ZIP (zip archive with compression) | <p>If you do not pass the <code>-a</code> 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.</p> |

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

| Option | Purpose | Key Considerations |
|---|---|--|
| <p><code>-f path_name</code></p> | <p>Specifies the path and file name of the snapshot to be created.</p> | <p>If you do not specify the <code>-f</code> option, the default file name of the server snapshot is:</p> <p><code>server_name_yymmdd-hhmm.tar.gz</code></p> <p>The <code>server_name</code> is the name of the server. The <code>yymmdd-hhmm</code> is the time the snapshot is taken.</p> <p>The default file name of the data snapshot is:</p> <p><code>server_name_yymmdd-hhmm_db.tar.gz</code></p> <p>The archive and compression selection set by the value in the <code>-a</code> option overrides any archive and compression settings indicated in the file name suffix set with the <code>-f</code> option. For example, if a command contains the following: <code>ifxdeployassist -a TAR -f C:\Informix\server.zip</code> then the snapshot will be saved as a tar archive with no compression rather than as a zip archive with compression.</p> |
| <p><code>-i connectivity_information</code></p> | <p>Specifies the connectivity information for the template instance from which to create a snapshot. If you do not use the <code>-i</code> option, the deployment assistant detects the connectivity information from the current environment and uses the same connectivity settings.</p> <p>In the place of <code>connectivity_information</code>, indicate specific connectivity settings for the snapshot to use after it is deployed. Set values for the following variables, separated by colons without spaces:</p> <ul style="list-style-type: none"> • 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 | <p>The only valid values for the protocol information are <code>SQLI</code> and <code>DRDA</code>. If no protocol is specified, the command sets the connection to <code>SQLI</code>.</p> <p>The connectivity information here is contained in the <code>sqlhosts</code> file.</p> <p>Example: <code>ifxdeployassist -i hrapps:oscar:40:DRDA</code></p> |
| <p><code>-c</code></p> | <p>Directs the deployment assistant to run in a command-line interface.</p> | <p>The <code>-c</code> 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.</p> |

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

| Option | Purpose | Key Considerations |
|--------|---|--|
| -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 <code>ifxdeployassist</code> command. | |
| -v | Runs the <code>ifxdeployassist</code> 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 file.
 - 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.

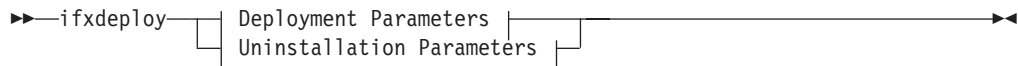
To deploy the snapshot on the target computer with a configuration file:

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.
5. Manually add the IBM Informix .NET drivers to the Global Assembly Cache (GAC).

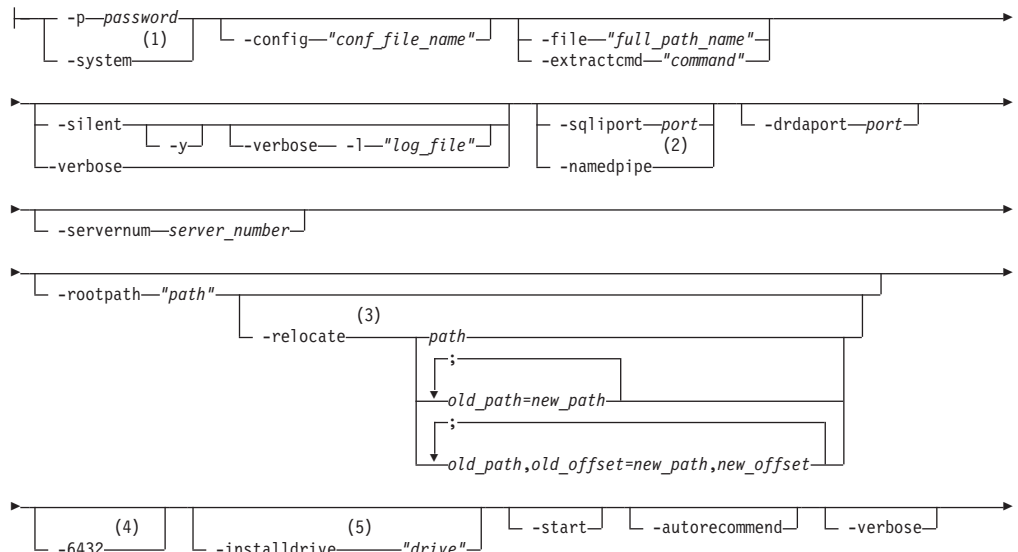
The ifxdeploy command: The deployment utility

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

Syntax



Deployment parameters:



┌ -force ─┐ ┌ -l—"log_file" ─┐ ┌ -y ─┐

Uninstallation parameters:

┌ -uninstall—"installation_path" ─┐ ┌ -delifx ─┐ ┌ (6) ─┐
└ -6432 ─┘

┌ -silent ─┐ ┌ -y ─┐ ┌ -verbose ─┐ ┌ -l—"log_file" ─┐
└ -verbose ─┘

┌ -y ─┐

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. |

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

| Element | Purpose | Key considerations |
|-------------------------------------|--|--|
| <code>-file "full_path_name"</code> | Extracts a compressed snapshot <i>only if</i> the file specified in <code>"full_path_name"</code> is in BZIP2, GZIP, TAR, or ZIP format. | <p>Windows: Indicate the location of the compressed image as an absolute path. Example: <code>-file C:\IBM\informix\my_archive.zip</code></p> <p>Linux and UNIX: Example: <code>-file /opt/IBM/informix/ifxdir.tgz</code></p> <p>If the snapshot is not in the BZIP2, GZIP, TAR, or ZIP compression format, use the <code>-extractcmd</code> option instead.</p> |
| <code>-extractcmd "command"</code> | Extracts a compressed snapshot, regardless of compression format. This option primarily is intended for when you want to extract a snapshot that is <i>not</i> in BZIP2, GZIP, TAR, or ZIP format or when you have specialized extraction procedure. | <p>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 <code>-file</code> option instead.</p> <p>Example on Windows: <code>-extractcmd "tar -xf C:\my_archive.tar c:\informix"</code></p> <p>Example on Linux and UNIX: <code>-extractcmd " (cd /placetoextract ; myextract /opt/IBM/informix/ifxdir) "</code></p> |
| <code>-force</code> | Overwrites the existing instance settings on the target computer. | |
| <code>-installdrive "drive"</code> | 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. |
| <code>-l "file_path"</code> | 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. |
| <code>-namedpipe</code> | This option is for Windows only. Sets the default database server network protocol to a named pipe connection. | If you specify both <code>-namedpipe</code> and <code>-sqlport</code> options, the deployment utility uses only one of them. |
| <code>-p password</code> | 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. |

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

| Element | Purpose | Key considerations |
|---------------------------------------|---|---|
| <code>-relocate path</code> | Specifies new parent directory of chunks in the deployed instance. You must have the appropriate ownership and permissions to modify the directories. | <p>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.</p> <p>Windows: If the parent directories have spaces in the paths, place the string of paths after the <code>-relocate</code> option in double quotation marks. Example:<code>ifxdeploy -relocate "C:\Program Files\IBM\IBM Informix\11.50\tmp=D:\IFMXDATA\myserver"</code></p> <p>Linux and UNIX: If multiple parent directories are specified and have spaces in the paths, place the string of paths after the <code>-relocate</code> option in single quotation marks. Example:<code>ifxdeploy -relocate '/opt/IBM/informix1150/tmp=/local/data/myserver;/opt/IBM/informix1150/dbspace=/local2/data/myserver'</code></p> |
| <code>-rootpath "path"</code> | Indicates the location of the root dbspace. | <p>Windows: Default path:<code>drive:\ifmxdata\server_name\rootdbs_dat.000</code></p> <p>Linux and UNIX: Default path:<code>\$INFORMIXDIR/server_name/rootdbs</code></p> |
| <code>-servernum server_number</code> | Specifies the server number of the instance. | The number must be an integer from 0 to 255. The default is 0. |
| <code>-silent</code> | 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 <code>-silent</code> option and <code>-verbose</code> option together, you must also include the <code>-l</code> argument. |
| <code>-sqliport port</code> | Specifies the SQLHOSTS service port for the server instance. | <p>The default port is 9088.</p> <p>If you specify both <code>-namedpipe</code> and <code>-sqliport</code>, the deployment utility uses only one of them.</p> |

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

| Element | Purpose | Key considerations |
|---|---|--|
| <code>-start</code> | 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 <code>-start</code> option to function. If you do not pass a value with the <code>-start</code> 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. |
| <code>-system</code> | This option is for Windows only. Creates an Informix service that logs on to the operating system as system user. | |
| <code>-uninstall installation_path</code> | Removes a snapshot that was originally deployed by the deployment utility. | |
| <code>-verbose</code> | Runs the command in verbose mode. | |
| <code>-y</code> | Runs the command without prompting for confirmation. | |
| <code>-6432</code> | 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 2-7 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 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 -sqlport 9090  
-servername 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 -sqlport 9090  
-drdport 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 -sqlport 9093 -drdport 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/`.

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 2-2 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 |
| PROTOCOL1 | Primary network protocol. This parameter is equivalent to the connection type in the <code>sqlhosts</code> information. | PROTOCOL1 olscotcp |
| SQLIPORT | SQLHOSTS service port for the server instance (not required for <code>onipcnp</code>). 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 <code>SERVERNUM</code> configuration parameter. The range of permissible values is from 0 to 255. | SERVERNUM 100 |

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

| Parameter | Description | Example of value setting (uncommented line) |
|-----------------------------|--|--|
| 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 |
| 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. | |

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

| Parameter | Description | Example of value setting (uncommented line) |
|---|--|---|
| 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. | <p>Example 1: Relocate all chunk paths to one directory:</p> <p>Windows: C:\example1</p> <p>UNIX and Linux: /example1</p> <p>Example 2: Select individual chunk paths and specify the directories where the paths are relocated:</p> <p>Windows: C:\ex2=C:\ex3; C:\ex4=C:\ex5</p> <p>UNIX and Linux: /ex2=/ex3;/ex4=C:ex5</p> <p>Example 3: Change multiple chunk paths and offsets:</p> <p>Windows: C:\ex6,10=C:\ex7,100; C:\ex8,20=C:\ex9,200</p> <p>UNIX and Linux: /ex6,10=/ex7,100; /ex8,20=/ex9,200</p> |
| 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 <i>password</i> |
| SYSTEM | Specifies whether the deployed instance logs on to Windows as a local system user or as user informix . This parameter is for Windows only. | <ul style="list-style-type: none"> • 0 (default). Informix logs on as user informix. • 1. Informix logs on as the Local System user and adds the current user to the Informix-Admin group. |
| LOGFILE | Sets the full path name for the log file of the deployment utility's errors and messages. | <p>Windows: LOGFILE c:\my_log.txt</p> <p>UNIX andLinux: LOGFILE /tmp/mylog</p> |
| LOGLEVEL | Sets amount of information to write to log. See the <i>ifxdeploy.conf</i> file for permissible values. | LOGLEVEL 5 |
| SILENT | Sets whether the utility displays console output while it is running. | <p>SILENT 1 (no console output)</p> <p>SILENT 0 (displays console output)</p> |
| FORCE | Overwrites existing environment variable and <i>onconfig</i> file settings of the target computer. | <p>FORCE 0 (Does not overwrite)</p> <p>FORCE 1 (Overwrites existing settings)</p> |
| INSTALLDRIVE This parameter is for Windows only. | Specifies the directory in which the deployed instance's dbspaces will be created on Windows. | INSTALLDRIVE C |

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

| Parameter | Description | Example of value setting (uncommented line) |
|--|---|--|
| ROOTPATH | Sets the location of the root dbspace. | Windows: Default path: <i>drive:\ifmxdata\server_name\rootdbs_dat.000</i> UNIX and Linux Default path: <i>\$INFORMIXDIR/server_name/rootdbs</i> |
| 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) |
|---|---|---|
| <p>BEGIN AUTORECOMMEND . . . END AUTORECOMMEND</p> | <p>Specifies the following parameters that are used to generate recommended values in the <i>onconfig</i> file:</p> <p>MAXCPUS Maximum number of processors and cores that the database server can use. Default: 1 Minimum value: 1 Maximum value: none</p> <p>MAXDISK Maximum amount of disk space that the database server can use. Default: 2048 MB Minimum value: 256 MB Maximum value: none</p> <p>MAXMEM Maximum amount of memory that the database server can use. Default: 512 MB</p> <p>MAXUSERS Maximum number of expected online transaction processing (OLTP) application users. Default: 32</p> <p>MAXDSUSERS Maximum number of expected decision-support system (DSS) application users. Default: 4</p> <p>RTO_SERVER_RESTART Recovery time objective for a server restart. Default: 60 seconds Minimum value: 60 seconds Maximum value: 1800 seconds</p> | <pre>BEGIN AUTORECOMMEND MAXCPUS 1 MAXDISK 2048 MAXMEM 512 MAXUSERS 32 MAXDSUSERS 4 RTO_SERVER_RESTART 60 END AUTORECOMMEND</pre> |
| <p>CLONE</p> | <p>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 <i>ifxdeploy.conf</i> file.</p> | <p>1</p> |
| <p>SOURCESERVER</p> | <p>Specifies the name of the source server.</p> | <p>SOURCESERVER clone 3</p> |

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 <i>password</i> |

Related concepts:

 The sqlhosts information (Administrator's Guide)

The syncsqlhosts utility

Use the **syncsqlhosts** utility to manually convert the connection information between the sqlhosts file format and the Windows registry format.

Syntax

```

>>> syncsqlhosts [-source—sqlhostvalue] [-target—sqlhostvalue] [-servers—serverlist]
  
```

Command options

Table 2-4. Options for the syncsqlhosts command

| Option | Purpose | Key considerations |
|-----------------------------|---|--|
| -source <i>sqlhostvalue</i> | Specifies the source machine to copy connection information from. <i>sqlhostvalue</i> is either a \\machine name or an sqlhosts file. | If you do not specify a -source option, the local registry is used. |

Table 2-4. Options for the syncsqlhosts command (continued)

| Option | Purpose | Key considerations |
|-----------------------------|---|--|
| -target <i>sqlhostvalue</i> | Specifies the target machine to copy connection information to. <i>sqlhostvalue</i> is either a \\machine name or an sqlhosts file. | If you do not specify a -target option, the local registry is used. |
| -servers <i>serverlist</i> | Specifies the servers to copy connection information from. <i>serverlist</i> is a list of Informix Server entries that are separated by spaces that are copied from the source to the target machine. | If you do not specify a -servers option, all of the defined connections are copied. |

Usage

The following prerequisites must be met to run this command:

- The user must have the appropriate authorization to read or write files on the source and target system and registry.
- If the specified value for a parameter contains a space, you must enclose that value in double quotation marks ("").

Examples of syncsqlhosts utility usage

In the following examples, you are issuing commands from a machine named machine1.

- To copy the connection information from the registry on machine2 to the registry on machine1:

```
syncsqlhosts -source \\machine2
```
- To copy the connection information from the registry on machine1 to the registry on machine3:

```
syncsqlhosts -source \\machine1 -target \\machine3
```
- To copy connection information from the registry on machine2 about servers ol_server1 and ol_server1test to the registry on machine1:

```
syncsqlhosts -source \\machine2 -servers ol_server1 ol_server1test
```
- To copy connection information from the registry on machine2 about servers ol_server1 and ol_server1test to a file on machine1:

```
syncsqlhosts -source \\machine2 -target c:\informix\etc\sqlhosts.ol_machine2  
-servers ol_machine1 ol_machine1test
```

Related concepts:

"Synchronize connectivity information between servers and clients on Windows" on page 1-2

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 root 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 root 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 that is repeated in multiple environments.

When you create 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 full product-version identifier and the architecture.

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 complete a deployment. If you uninstall the product, you need to use the image file name .

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:

1. Be sure you are logged in with root-level privileges.

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.

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

```
rpm -e informix-release_identifier-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.

Related concepts:

“Planning for deploying Informix” on page 2-2

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
```

Related tasks:

“Creating a snapshot with the deployment assistant” on page 2-2

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 root path

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

Related tasks:

Chapter 2, “Deploying Informix software,” on page 2-1

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 12.10 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.

Related concepts:

 [Installing Informix on UNIX, Linux, and Mac OS X \(Installation Guide \(UNIX\)\)](#)

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.

Related concepts:

“Easy deployment” on page 1-2

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.
 - 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.

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 12.10 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.

Related concepts:

 [Installing Informix on Windows \(Installation Guide \(Windows\)\)](#)

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 **sysadmin** 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.
 - 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.

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 row and simple large objects in dbspaces, consolidate the remaining free space, and return the free space to the dbspace.
- Compress only row data or only simple large objects in dbspaces.
- Compress detached B-tree indexes, consolidate free space in the index, and return the free space to the dbspace. You can also compress indexes when you use create them with the CREATE INDEX statement.
- Implement automatic space management
- Defragment partitions to non-contiguous extents

Related concepts:

- [➤ Compression \(Administrator's Guide\)](#)
- [➤ B-tree index compression \(Administrator's Guide\)](#)
- [➤ Automatic space management \(Administrator's Guide\)](#)
- [➤ Partition defragmentation \(Administrator's Guide\)](#)

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. |

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

| Message log file | Task that rotates the file | Default frequency for when task runs |
|------------------|----------------------------|--|
| 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 log 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.

To manually rotate a message log file:

1. 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.log` files, specify:

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

To manually truncate a message log file (remove the contents of the file):

1. 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");
```

To manually delete a message log file (remove the contents of the file and the file):

1. 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");
```

Related reference:

[message log rotate argument](#): Rotate the message log file (SQL administration API) (Administrator's Reference)

[message log truncate argument](#): Delete the contents of a message log file (SQL administration API) (Administrator's Reference)

[message log delete argument](#): Delete a message log file (SQL administration API) (Administrator's Reference)

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.

Related reference:

[LOW_MEMORY_RESERVE configuration parameter](#) (Administrator's Reference)

[Reserve memory for critical activities](#) (Administrator's Guide)

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.

Related concepts:

[🔗](#) Configure the server response when memory is critically low (Administrator's Guide)

Related reference:

[🔗](#) LOW_MEMORY_MGR configuration parameter (Administrator's Reference)

[🔗](#) scheduler lmm enable argument: Specify automatic low memory management settings (SQL administration API) (Administrator's Reference)

[🔗](#) onstat -g lmm command: Print low memory management information (Administrator's Reference)

Appendix. Accessibility

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

Accessibility features for IBM Informix products

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

Accessibility features

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

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

Keyboard navigation

This product uses standard Microsoft Windows navigation keys.

Related accessibility information

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

IBM and accessibility

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

Dotted decimal syntax diagrams

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

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

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

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

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

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

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

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

- ? Specifies an optional syntax element. A dotted decimal number followed by the ? symbol indicates that all the syntax elements with a corresponding dotted decimal number, and any subordinate syntax elements, are optional. If there is only one syntax element with a dotted decimal number, the ? symbol is displayed on the same line as the syntax element (for example, 5? NOTIFY). If there is more than one syntax element with a dotted decimal number, the ? symbol is displayed on a line by itself, followed by the syntax elements that are optional. For example, if you hear the lines 5 ?, 5 NOTIFY, and 5 UPDATE, you know that syntax elements NOTIFY and UPDATE are optional; that is, you can choose one or none of them. The ? symbol is equivalent to a bypass line in a railroad diagram.
- ! Specifies a default syntax element. A dotted decimal number followed by the ! symbol and a syntax element indicates that the syntax element is the default option for all syntax elements that share the same dotted decimal number. Only one of the syntax elements that share the same dotted decimal number can specify a ! symbol. For example, if you hear the lines 2? FILE, 2.1! (KEEP), and 2.1 (DELETE), you know that (KEEP) is the default option for the FILE keyword. In this example, if you include the FILE keyword but do not specify an option, default option KEEP is applied. A default option also applies to the next higher dotted decimal number. In this example, if the FILE keyword is omitted, default FILE(KEEP) is used. However, if you hear the lines 2? FILE, 2.1, 2.1.1! (KEEP), and 2.1.1 (DELETE), the default option KEEP only applies to the next higher dotted decimal number, 2.1 (which does not have an associated keyword), and does not apply to 2? FILE. Nothing is used if the keyword FILE is omitted.
- * Specifies a syntax element that can be repeated zero or more times. A dotted decimal number followed by the * symbol indicates that this syntax element can be used zero or more times; that is, it is optional and can be

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

Notes:

1. If a dotted decimal number has an asterisk (*) next to it and there is only one item with that dotted decimal number, you can repeat that same item more than once.
 2. If a dotted decimal number has an asterisk next to it and several items have that dotted decimal number, you can use more than one item from the list, but you cannot use the items more than once each. In the previous example, you can write HOST STATE, but you cannot write HOST HOST.
 3. The * symbol is equivalent to a loop-back line in a railroad syntax diagram.
- + Specifies a syntax element that must be included one or more times. A dotted decimal number followed by the + symbol indicates that this syntax element must be included one or more times. For example, if you hear the line 6.1+ data-area, you must include at least one data area. If you hear the lines 2+, 2 HOST, and 2 STATE, you know that you must include HOST, STATE, or both. As for the * symbol, you can repeat a particular item if it is the only item with that dotted decimal number. The + symbol, like the * symbol, is equivalent to a loop-back line in a railroad syntax diagram.

Notices

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

COPYRIGHT LICENSE:

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

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

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

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

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

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

Privacy policy considerations

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

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

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

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

Trademarks

IBM, the IBM logo, and [ibm.com](http://www.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

A

- Accessibility A-1
 - dotted decimal format of syntax diagrams A-1
 - keyboard A-1
 - shortcut keys A-1
 - syntax diagrams, reading in a screen reader A-1
- Administration
 - automating 1-3

C

- changing the frequency 4-3
- client connection
 - synchronizing with server 1-2
- clone 2-1
- Command line
 - deploying Informix snapshot 2-8
- compliance with standards vi
- Configuration 1-3
- Configuration parameters
 - LOW_MEMORY_MGR configuration parameter 4-5
 - LOW_MEMORY_RESERVE configuration parameter 4-5

D

- Deploying
 - an instance 1-2
- Deploying Informix 2-2
- deploying to multiple computers 2-1
- Deployment
 - planning 2-2
- deployment assistant 3-1
 - command 2-4
 - creating a snapshot 2-3
- Deployment assistant 3-1
- deployment utility 2-1, 3-2
 - removing a snapshot 2-24
 - syntax and options 2-8
 - using with ifxdeploy.conf file 2-16
- Deployment utility 2-2
- Disabilities, visual
 - reading syntax diagrams A-1
- Disability A-1
- disabling rotation tasks 4-2
- Dotted decimal format of syntax diagrams A-1

E

- Embeddability Toolkit
 - components 3-1
 - create a snapshot on Linux 3-4
 - create a snapshot on Windows 3-8
 - deploy snapshots on Linux 3-5
 - deploy snapshots on Windows 3-9
 - IBM Informix snapshots
 - deploying on Linux 3-5
 - install on Linux 3-4
 - install on Windows 3-7
 - tutorial 3-1, 3-4, 3-5, 3-7, 3-8, 3-9

- Embedded applications
 - maintaining targeted memory 4-5
 - reserving memory 4-5
- enabling rotation tasks 4-2
- Environment 1-4

H

- High-availability overview 1-5

I

- IBM Informix snapshots
 - deploying on Windows 3-9
- ifxdeploy command 2-7
 - examples on UNIX 2-8
 - examples on Windows 2-8
 - syntax 2-8
 - usage 2-8
- ifxdeploy command options
 - 6432 2-8
 - autorecomend 2-8
 - config 2-8
 - delifx 2-8
 - drdaport 2-8
 - extractcmd 2-8
 - file 2-8
 - force 2-8
 - installdrive 2-8
 - l 2-8
 - namedpipe 2-8
 - p 2-8
 - relocate 2-8
 - rootpath 2-8
 - servernum 2-8
 - silent 2-8
 - sqlipport 2-8
 - start 2-8
 - system 2-8
 - uninstall 2-8
 - verbose 2-8
 - y 2-8
- ifxdeploy utility 3-2
- ifxdeploy.conf file
 - parameters 2-16
 - purpose 2-16
- ifxdeploy.conf file parameters
 - BEGIN ALIAS . . . END ALIAS 2-16
 - BEGIN AUTORECOMMEND . . . END AUTORECOMMEND
 - MAXCPU 2-16
 - MAXDISK 2-16
 - MAXDSUSERS 2-16
 - MAXMEM 2-16
 - MAXUSERS 2-16
 - RTO_SERVER_RESTART 2-16
 - BEGIN ONCONFIG . . . END ONCONFIG 2-16
 - CLONE 2-16
 - CLONEIPADDR 2-16
 - DISPOSITION 2-16

ifxdeploy.conf file parameters (continued)

- DRDAPORT 2-16
- FORCE 2-16
- INFORMIXDIR 2-16
- INFORMIXPASSWORD 2-16
- INFORMIXSERVER 2-16
- INFORMIXSQLHOSTS 2-16
- INSTALLDRIVE 2-16
- LOGFILE 2-16
- LOGLEVEL 2-16
- ONCONFIG 2-16
- PASSWORD 2-16
- PORT1 2-16
- PROTOCOL1 2-16
- RELOCATE 2-16
- ROOTPATH 2-16
- SERVERNUM 2-16
- SILENT 2-16
- SNAPSHOT 2-16
- SOURCEIPADDR 2-16
- SOURCEPORT 2-16
- SOURCESERVER 2-16
- SQLIPORT 2-16
- START 2-16
- SYSTEM 2-16
- TARGETSIZE 2-16
- TRUSTED 2-16
- USELOCAL 2-16
- USERNAME 2-16
- WIN6432 2-16

ifxdeployassist command 2-4, 3-1

industry standards vi

Installation

- customizing 1-1

installing

- using RPM Package Manager 2-24

L

LOW_MEMORY_MGR configuration parameter 4-5

LOW_MEMORY_RESERVE configuration parameter 4-5

M

managing 1-3, 4-1, 4-2

Memory 4-1

- reserving 4-5

- targeted amount 4-5

Message logs 4-1, 4-2

- changing the threshold 4-4

- managing 4-2, 4-3, 4-4

- rotation 4-2, 4-3, 4-4

O

onconfig file

- specifying environment 1-4

R

Response file 1-1

RPM Package Manager 2-24

S

Screen reader

- reading syntax diagrams A-1

server connection

- synchronizing with client 1-2

Shortcut keys

- keyboard A-1

silent installation 2-24

Silent installation

- customizing 1-1

snapshot 2-1

- creating for deployment 2-3, 2-7

- deploying 2-1, 2-7

- removing 2-24

specifying in onconfig file 1-4

sqlhosts file 1-2

standards vi

sync_registry task

- about 1-2

syncsqlhosts utility

- about 1-2

- options 2-22

- syntax 2-22

- usage 2-22

Syntax diagrams

- reading in a screen reader A-1

U

uninstalling

- using RPM Package Manager 2-24

Utilities

- syncsqlhosts 2-22

- syncsqlhosts syntax 2-22

V

Visual disabilities

- reading syntax diagrams A-1



Printed in USA

SC27-4519-01

