



BEA WebLogic Integration™

Migrating to BEA WebLogic Integration Release 2.1

Copyright

Copyright © 2002 BEA Systems, Inc. All Rights Reserved.

Restricted Rights Legend

This software and documentation is subject to and made available only pursuant to the terms of the BEA Systems License Agreement and may be used or copied only in accordance with the terms of that agreement. It is against the law to copy the software except as specifically allowed in the agreement. This document may not, in whole or in part, be copied photocopied, reproduced, translated, or reduced to any electronic medium or machine readable form without prior consent, in writing, from BEA Systems, Inc.

Use, duplication or disclosure by the U.S. Government is subject to restrictions set forth in the BEA Systems License Agreement and in subparagraph (c)(1) of the Commercial Computer Software-Restricted Rights Clause at FAR 52.227-19; subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013, subparagraph (d) of the Commercial Computer Software--Licensing clause at NASA FAR supplement 16-52.227-86; or their equivalent.

Information in this document is subject to change without notice and does not represent a commitment on the part of BEA Systems. THE SOFTWARE AND DOCUMENTATION ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND INCLUDING WITHOUT LIMITATION, ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. FURTHER, BEA Systems DOES NOT WARRANT, GUARANTEE, OR MAKE ANY REPRESENTATIONS REGARDING THE USE, OR THE RESULTS OF THE USE, OF THE SOFTWARE OR WRITTEN MATERIAL IN TERMS OF CORRECTNESS, ACCURACY, RELIABILITY, OR OTHERWISE.

Trademarks or Service Marks

BEA, Jolt, Tuxedo, and WebLogic are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Campaign Manager for WebLogic, BEA eLink, BEA Manager, BEA WebLogic Commerce Server, BEA WebLogic E-Business Platform, BEA WebLogic Enterprise, BEA WebLogic Express, BEA WebLogic Integration, BEA WebLogic Personalization Server, BEA WebLogic Portal, BEA WebLogic Server and How Business Becomes E-Business are trademarks of BEA Systems, Inc.

All other trademarks are the property of their respective companies.

Migrating to BEA WebLogic Integration Release 2.1

Part Number	Date	Software Version
N/A	January 2002	2.1 Service Pack 1

Contents

About This Document

What You Need to Know	viii
e-docs Web Site	viii
How to Print the Document	viii
Related Information	ix
Contact Us!	ix
Documentation Conventions	x

1. Introduction to Migration

Migration Overview	1-1
Why Migrate?	1-2
Migration Procedures	1-2

2. Migrating from WebLogic Process Integrator 1.2 or 1.2.1

Before You Begin	2-1
Migration Process Overview	2-2
Migrating the Database	2-2
Migrating the Database Schema	2-3
Migrating the Security Realm	2-4
Select a Migration Procedure	2-5
Migrate from the RDBMS Realm (WebLogic Process Integrator 1.2 or 1.2.1)	2-5
Migrate from the File Realm (WebLogic Process Integrator 1.2.1 Only)	2-8
Assigning Permissions	2-9

3. Migrating from WebLogic Collaborate 1.0 or 1.0.1

Before You Begin	3-1
Migration Process Overview	3-2
Migrating the Repository and Workflows	3-3
What Is Migrated?	3-3
Repository Migration and Workflow Conversion Procedure.....	3-3
Migrating Applications to the WebLogic Integration 2.1 Messaging API.....	3-7
Introduction to Application Migration	3-8
Importing XOCP Applications	3-9
Updating the Terminology	3-9
Supporting the C-Enabler API for WebLogic Collaborate 1.0 or 1.0.1 ...	3-13
C-Enabler Configuration	3-13
Repository	3-14
WebLogic Collaborate 1.0 or 1.0.1 C-Enabler API Implementation	3-14
Required Packages	3-14
Required Interfaces	3-14
Required Classes	3-15

4. Migrating from WebLogic Integration 2.0 to WebLogic Integration 2.1 Service Pack 1

Migrating from WebLogic Integration 2.0 with No Service Pack or Service Pack 1 to WebLogic Integration 2.1 Service Pack 1.....	4-2
Migrating from WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to WebLogic Integration 2.1 Service Pack 1	4-3
Migration Process Overview	4-3
Before You Begin.....	4-4
Domain Migration.....	4-4
Remote Database Server	4-5
Security File Realms	4-5
Database Migration	4-5

5. Migrating WebLogic Integration 2.1 to WebLogic Integration 2.1 Service Pack 1

Before You Begin	5-1
Domain Migration	5-2

Database Migration	5-3
Upgrade Install	5-5
New Installation	5-7
Migrating Two Repositories.....	5-7

6. Migration Utilities

Migrating with the Database Configuration Wizard	6-1
Migrating with the Command-Line Migration Script	6-7
Windows	6-7
UNIX.....	6-7



About This Document

Migrating to BEA WebLogic Integration Release 2.1 is organized as follows:

- Chapter 1, “Introduction to Migration,” provides background on migrating to WebLogic Integration 2.1 Service Pack 1 (SP1).
- Chapter 2, “Migrating from WebLogic Process Integrator 1.2 or 1.2.1,” describes the procedures for migrating WebLogic Process Integrator 1.2 or 1.2.1 to BEA WebLogic Integration 2.1 SP1.
- Chapter 3, “Migrating from WebLogic Collaborate 1.0 or 1.0.1,” describes the procedures for migrating WebLogic Collaborate 1.0 or 1.0.1 to BEA WebLogic Integration 2.1 SP1.
- Chapter 4, “Migrating from WebLogic Integration 2.0 to WebLogic Integration 2.1 Service Pack 1,” describes the procedures for migrating BEA WebLogic Integration 2.0 to BEA WebLogic Integration 2.1 SP1.
- Chapter 5, “Migrating WebLogic Integration 2.1 to WebLogic Integration 2.1 Service Pack 1,” describes the procedures for migrating from BEA WebLogic Integration 2.1 to BEA WebLogic Integration 2.1 SP1.
- Chapter 6, “Migration Utilities,” describes the procedures for migrating with the WebLogic Integration database configuration wizard or with the command-line migration script

What You Need to Know

Migrating to BEA WebLogic Integration Release 2.1 is designed for WebLogic Integration users who want to migrate previous versions of the product to WebLogic Integration 2.1 Service Pack 1. Specifically, it provides instructions for migrating systems from the following products:

- WebLogic Process Integrator 1.2 or 1.2.1
- WebLogic Collaborate 1.0 or 1.0.1
- WebLogic Integration 2.0
- WebLogic Integration 2.1

e-docs Web Site

BEA product documentation is available on the BEA corporate Web site. From the BEA Home page, click on Product Documentation or go directly to the “e-docs” Product Documentation page at <http://edocs.bea.com>.

How to Print the Document

You can print a copy of this document from a Web browser, one file at a time, by using the File→Print option on your Web browser.

A PDF version of this document is available on the WebLogic Integration documentation Home page on the e-docs Web site (and also on the documentation CD). You can open the PDF in Adobe Acrobat Reader and print the entire document (or a portion of it) in book format. To access the PDFs, open the WebLogic Integration documentation Home page, click the PDF files button and select the document you want to print.

If you do not have the Adobe Acrobat Reader, you can get it for free from the Adobe Web site at <http://www.adobe.com/>.

Related Information

The following resources are also available:

- BEA WebLogic Server documentation (<http://e-docs.beasys.com>)
- BEA WebLogic Process Integrator documentation (<http://e-docs.beasys.com>)
- BEA WebLogic Collaborate documentation (<http://e-docs.beasys.com>)
- BEA WebLogic Integration 2.1 documentation (<http://e-docs.beasys.com>)

Contact Us!

Your feedback on the WebLogic Integration documentation is important to us. Send us e-mail at **docsupport@beasys.com** if you have questions or comments. Your comments will be reviewed directly by the BEA professionals who create and update the WebLogic Integration documentation.

In your e-mail message, please indicate that you are using the documentation for this release of WebLogic Integration.

If you have any questions about this version of WebLogic Integration, or if you have problems installing and running WebLogic Integration, contact BEA Customer Support through BEA WebSUPPORT at **www.beasys.com**. You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

When contacting Customer Support, be prepared to provide the following information:

- Your name, e-mail address, phone number, and fax number

- Your company name and company address
- Your machine type and authorization codes
- The name and version of the product you are using
- A description of the problem and the content of pertinent error messages

Documentation Conventions

The following documentation conventions are used throughout this document.

Convention	Item
Ctrl+Tab	Indicates that you must press two or more keys simultaneously.
<i>italics</i>	Indicates emphasis or book titles.
monospace text	Indicates code samples, commands and their options, data structures and their members, data types, directories, and file names and their extensions. Monospace text also indicates text that you must enter from the keyboard. <i>Examples:</i> <pre>#include <iostream.h> void main () the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float</pre>
monospace boldface text	Identifies significant words in code. <i>Example:</i> <pre>void commit ()</pre>
<i>monospace italic text</i>	Identifies variables in code. <i>Example:</i> <pre>String <i>expr</i></pre>

Convention	Item
UPPERCASE TEXT	Indicates device names, environment variables, and logical operators. <i>Examples:</i> LPT1 SIGNON OR
{ }	Indicates a set of choices in a syntax line. The braces themselves should never be typed.
[]	Indicates optional items in a syntax line. The brackets themselves should never be typed. <i>Example:</i> buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...
	Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.
...	Indicates one of the following in a command line: <ul style="list-style-type: none"> ■ That an argument can be repeated several times in a command line ■ That the statement omits additional optional arguments ■ That you can enter additional parameters, values, or other information The ellipsis itself should never be typed. <i>Example:</i> buildobjclient [-v] [-o name] [-f file-list]... [-l file-list]...
.	Indicates the omission of items from a code example or from a syntax line. The vertical ellipsis itself should never be typed.



1 Introduction to Migration

This document provides the procedures you must complete to migrate an installation of WebLogic Collaborate 1.0 or 1.0.1, WebLogic Process Integrator 1.2 or 1.2.1, WebLogic Integration 2.0, or WebLogic Integration 2.1 to WebLogic Integration 2.1 Service Pack 1.

This section provides information about the following topics:

- Migration Overview
- Why Migrate?
- Migration Procedures

Migration Overview

WebLogic Integration provides the tools that businesses need to develop new applications, integrate new and existing systems, streamline business processes, and connect with trading partners. It makes it possible for an enterprise to integrate multiple applications of its own (intraenterprise integration) and to integrate its own applications with those of other enterprises (interenterprise integration). The WebLogic Integration tools support the following types of functionality:

- Application Integration
- Business Process Management (BPM)

- Business-to-Business Integration (B2B)
- Data Integration

Before the release of BEA WebLogic Integration 2.0, the business process management functionality was available in the WebLogic Process Integrator product, and the business-to-business integration functionality was available in the WebLogic Collaborate product.

Why Migrate?

- WebLogic Integration 2.1 Service Pack 1 (SP1) requires additional database entities not required for WebLogic Integration 2.1. For compatibility with WebLogic Integration 2.1 Service Pack 1, you must update your existing WebLogic Integration 2.1 repository with these additional database entities.
- BEA WebLogic Integration 2.1 uses a new database schema and security model. If you are using WebLogic Integration 2.0, WebLogic Process Integrator 1.2 or 1.2.1, or WebLogic Collaborate 1.0 or 1.0.1, you must migrate your existing database and security realm for compatibility with WebLogic Integration 2.1.

Migration Procedures

This document provides separate procedures for migrating to WebLogic Integration Service Pack 1 (SP1) from the following products:

- WebLogic Process Integrator 1.2 or 1.2.1
- WebLogic Collaborate 1.0 or 1.0.1
- WebLogic Integration 2.0 with no service pack or with Service Pack 1
- WebLogic Integration 2.0 with Service Pack 2, 3, or 4
- WebLogic Integration 2.1

2 Migrating from WebLogic Process Integrator 1.2 or 1.2.1

This section provides procedures for migrating an existing installation of WebLogic Process Integrator 1.2 or 1.2.1 to WebLogic Integration 2.1 Service Pack 1 (SP1). Specifically, it provides information about the following subjects:

- Before You Begin
- Migration Process Overview
- Migrating the Database

Before You Begin

The WebLogic Integration 2.1 Service Pack 1 (SP1) installation includes all the tools and files you need to migrate your databases and security realm. You do not need to have WebLogic Integration 2.0 installed to perform any migration procedure.

Before you attempt to migrate to WebLogic Integration 2.1 SP1, we strongly recommend that you back up your entire database and export all your workflows. With this preparation, you will be able to fix your environment and start the migration again if your migration attempt fails. You can run the migration script as many times as necessary without causing any adverse results.

Migration Process Overview

To migrate WebLogic Process Integrator 1.2 or 1.2.1 to WebLogic Integration 2.1 Service Pack 1 (SP1):

1. Migrate the database schema to the WebLogic Integration 2.1 SP1 format as described in “Migrating the Database Schema” on page 2-3.
2. Migrate the security realm, as described in “Migrating the Security Realm” on page 2-4. Migration of the security realm is required because WebLogic Integration 2.1 runs on a version of WebLogic Server other than that on which WebLogic Process Integrator 1.2 or 1.2.1 runs.

After completing your database migration and security realm migration, you must also assign permissions to existing users and roles. For more information, see “Assigning Permissions” on page 2-9.

Once the database is converted to the 2.1 level, tables that might not have existed in your WebLogic Process Integrator 1.2 or 1.2.1 database are added to WebLogic Integration. The resulting database contains the overall WebLogic Integration repository, instead of only the WebLogic Process Integrator tables.

Migrating the Database

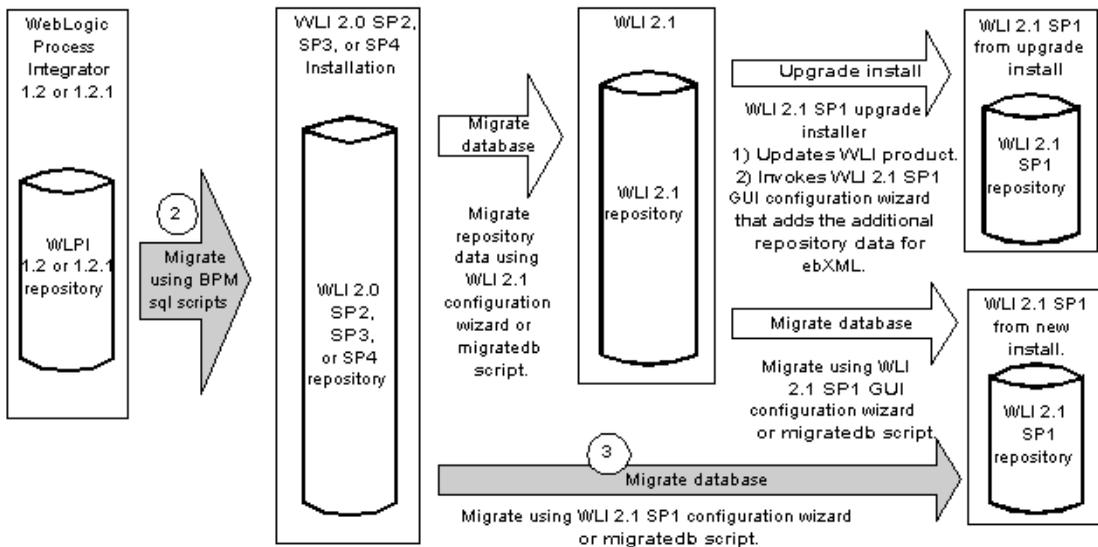
This section provides instructions for the following tasks:

- Migrating the Database Schema
- Migrating the Security Realm
- Assigning Permissions

Migrating the Database Schema

This section provides a procedure for converting the WebLogic Process Integrator 1.2 or 1.2.1 database schema to the WebLogic Integration 2.1 Service Pack 1 format, as represented in the following figure.

Figure 2-1 Migration from WebLogic Process Integrator 1.2 or 1.2.1 to WebLogic Integration 2.1 Service Pack 1



To migrate your database schema:

1. Using your database administration tool, back up your existing WebLogic Integration component database, as recommended in “Migrating the Database” on page 2-2.

2 Migrating from WebLogic Process Integrator 1.2 or 1.2.1

2. With the WebLogic Integration component database open, execute the appropriate `BPM_12x-20.sql` file listed in Table 2-1.

Table 2-1 WebLogic Process Integrator Database Migration Scripts

To migrate WebLogic Process Integrator tables for this database . . .	Use this dbscript file . . .
Oracle 8.1.6	<code>WLI_HOME/dbscripts/oracle/migrate/BPM_12x-20.sql</code>
Microsoft SQL Server 7.0 with Service Pack 3 or Microsoft SQL Server 2000	<code>WLI_HOME/dbscripts/mssql/migrate/BPM_12x-20.sql</code>
Sybase System 11.9.2	<code>WLI_HOME/dbscripts/sybase/migrate/BPM_12x-20.sql</code>

All files are located in `WLI_HOME/dbscripts/DB_TYPE/migrate`, where `DB_TYPE` is the subdirectory named for the type of database being migrated, such as Oracle. For directions on how to run the sql script for your database, see your database administrator.

3. Migrate the database schema from WebLogic Integration 2.0 format to WebLogic Integration 2.1 SP1 format by completing the steps in “Migrating from WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to WebLogic Integration 2.1 Service Pack 1” on page 4-3.

Now proceed to “Migrating the Security Realm” on page 2-4.

Migrating the Security Realm

WebLogic Integration 2.1 uses a new security model which you need to adopt in order to access Studio and Worklist functions. (For more information, see “[About Security Realms](#)” in *Using the WebLogic Integration Studio*.)

The `WLI_HOME/dbscripts` directory of your WebLogic Integration 2.1 installation includes a utility you can use to upgrade your security realm, along with an additional utility that allows you to run the tutorial example without any additional setup. (For more information, see [Learning to Use BPM with WebLogic Integration](#).)

Note: If you need to assign all permissions to existing users of your system, you can add the users to the list in the `WLPI_SUPER_USERS` variable, which is defined in the `Migration.properties` file. Otherwise you can assign permissions in the Studio client application after your migration is complete. For information about the default users and passwords provided in a new WebLogic Integration 2.1 installation, see “About Security Realms” in *Using the WebLogic Integration Studio*.

Select a Migration Procedure

Based on the release of WebLogic Process Integrator you are migrating and the security realm you have deployed, choose one of the migration procedures listed in the following table.

Table 2-2 Migration Procedures

If you are migrating from . . .	Do this . . .
WebLogic Process Integrator 1.2 or 1.2.1 and you deployed the <code>com.bea.wlpi.rdbmsrealm.RDBMSRealm</code> security realm	Use the procedure described in “Migrate from the RDBMS Realm (WebLogic Process Integrator 1.2 or 1.2.1)” on page 2-5.
WebLogic Process Integrator 1.2.1 and you deployed the File Realm security realm	Use the procedure described in “Migrate from the File Realm (WebLogic Process Integrator 1.2.1 Only)” on page 2-8.
WebLogic Process Integrator 1.2 and you deployed the <code>weblogic.Properties</code> security realm	There is no automatic migration path. Please contact BEA Customer Support for more information.

Migrate from the RDBMS Realm (WebLogic Process Integrator 1.2 or 1.2.1)

The following procedures are based on the assumption that you are familiar with WebLogic Server 6.1 and the WebLogic Server Administration Console. For more information, see the WebLogic Server documentation at the following URL:

<http://edocs-bea.com/wls/docs61/index.html>

2 Migrating from WebLogic Process Integrator 1.2 or 1.2.1

If you are migrating from WebLogic Process Integrator 1.2 or 1.2.1 and you have deployed the security realm `com.bea.wlpi.rdbmsrealm.RDBMSRealm`, complete the following procedure:

1. Start the WebLogic Server from the WebLogic Integration 2.1 BPM domain, as described in *Starting, Stopping, and Customizing BEA WebLogic Integration*.
2. Start the WebLogic Server Administration Console.
3. Choose Security→Realms→wlpiRDBMSRealm. In the right pane, select the Configuration and Schema tabs.
4. Back up the Schema Properties by copying the text to a temporary file.
5. In the migrate directory of your WebLogic Process Integrator installation, open the file `rdbmsrealm_migration_schema.txt` in a text editor.
6. Copy the text from the `rdbmsrealm_migration_schema.txt` file and paste it into the Schema Properties window in the WebLogic Server Administration Console, overwriting the original text. Click Apply.
7. In the right pane, select the Database tab.
8. Verify that the database URL in the JDBC connection pool specifies your WebLogic Process Integrator 1.2 or 1.2.1 database, and modify any settings that are incorrect for this database. If you make any modifications, restart the WebLogic Server.
9. Select Security in the left pane and select the Filerealm tab in the right pane.
10. From the right pane, select `wlpiCachingRealm` from the Caching Realm drop-down list.
11. In the left pane, choose Services→JDBC→Connection Pools→wliPool. In the right pane, select the Configuration and General tabs.
12. Verify that the database URL in the JDBC connection pool specifies your WebLogic Process Integrator 1.2 or 1.2.1 database, and modify any settings that are incorrect for this database.
13. Shut down the WebLogic Server.
14. From the `WLI_HOME/config/bpmdomain/` subdirectory of your WebLogic Integration 2.1 installation, open the startup script file, `startWeblogic.cmd` (on Windows) or `startWeblogic` (on UNIX), in a text editor.

15. To the `CLASSPATH` setting, immediately following `-classpath`, add the following:

```
%WLI_HOME%\lib\oldrdbmsrealm.jar;
```

The line should look as follows:

```
-classpath %WLI_HOME%\lib\oldrdbmsrealm.jar;%WLISERVERCP%
```

16. Save the file.
17. Start the WebLogic Server. Ignore any error messages you may receive.
18. Run the realm migration utility by going to the `WLI_HOME/dbscripts/` directory and doing one of the following:
 - If you are using Windows NT, run the `bpm20_realm_migrator.cmd` script.
 - If you are using UNIX, run `bpm20_realm_migrator`.
19. Start the WebLogic Server Administration Console.
20. Copy the text from the temporary file you created in step 4 and paste it into the schema properties in the WebLogic Server Administration Console, overwriting the original text. Click Apply.

Note: If you are using MSSQL7, you also need to change the schema properties by changing the setting of `getGroupNewStatement` from `false` to `true` (that is, `getGroupNewStatement=true`). For the other database types, the value `false` is correct.
21. Shut down the WebLogic Server.
22. From the server startup script, remove the text you added to the `CLASSPATH` setting in step 15, and save the file.
23. Start the WebLogic Server.
24. Optionally, run the tutorial enabler utility by going to the `WLI_HOME/dbscripts/` directory and doing one of the following:
 - If you are using Windows NT, run `bpm20_tutorial_enabler.cmd`.
 - If you are using UNIX, run `bpm20_tutorial_enabler`.

Migrate from the File Realm (WebLogic Process Integrator 1.2.1 Only)

The following procedures are based on the assumption that you are knowledgeable about WebLogic Server 6.1 and the WebLogic Server Administration Console. For more information, see the WebLogic Server documentation set at the following URL:

<http://edocs.bea.com/wls/docs61/index.html>

If you are migrating from WebLogic Process Integrator 1.2 or 1.2.1 and you deployed a file realm, complete the following steps:

1. Start the WebLogic Server from the BPM domain, as described in *Starting, Stopping, and Customizing BEA WebLogic Integration*.
2. Start the WebLogic Server Administration Console.
3. Choose Services→JDBC→Connection Pools→wliPool. In the right pane, select the Configuration and General tabs.
4. Verify that the database URL in the JDBC connection pool specifies your WebLogic Process Integrator 1.2.1 database and modify any settings that are incorrect for this database.
5. Shut down the WebLogic Server.
6. Merge the `fileRealm.properties` files from the WebLogic Process Integrator 1.2.1 domain into the `fileRealm.properties` file in the `WLI_HOME/config/bpmdomain` directory of your WebLogic Integration 2.1 installation.
7. Start the WebLogic Server. Ignore any messages you may receive.
8. Run the realm migration utility by going to the `WLI_HOME/dbscripts/` directory and doing one of the following:
 - If you are using Windows NT, run `bpm20_realm_migrator.cmd`.
 - If you are using UNIX, run `bpm20_realm_migrator`.
9. If you like, run the tutorial enabler utility by going to the `WLI_HOME/dbscripts/` directory and doing one of the following:
 - If you are using Windows NT, run `bpm20_tutorial_enabler.cmd`.
 - If you are using UNIX, run `bpm20_tutorial_enabler`.

Assigning Permissions

Once you have migrated your database and security realm, you must assign permissions to existing users and roles. You can do so with the WebLogic Integration Studio. For information about default permission groups and their member users, and about assigning permissions to users and roles, see [“Administering Data”](#) in *Using the WebLogic Integration Studio*.

3 Migrating from WebLogic Collaborate 1.0 or 1.0.1

This section provides procedures you can follow to migrate WebLogic Collaborate Release 1.0 or 1.0.1 to WebLogic Integration 2.1 Service Pack 1 (SP1). Specifically, it provides information about the following topics:

- Before You Begin
- Migration Process Overview
- Migrating the Repository and Workflows
- Migrating Applications to the WebLogic Integration 2.1 Messaging API

Before You Begin

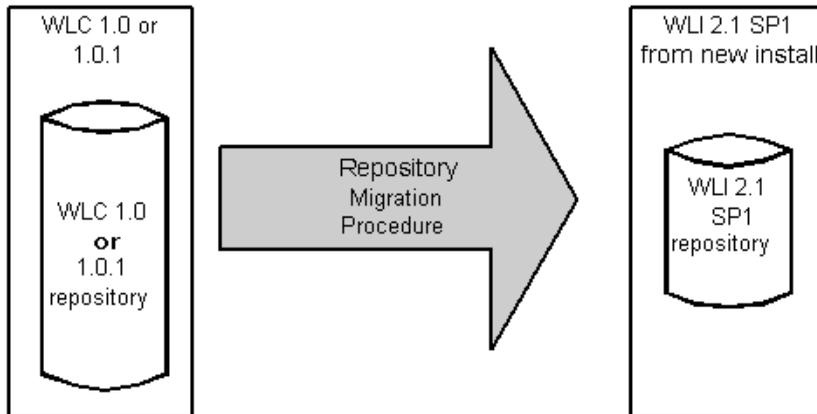
Before you attempt to migrate to WebLogic Integration 2.1 Service Pack 1, we strongly recommend that you back up your entire database and export all your workflows. With this preparation, you will be able to fix your environment and start the migration again if your first migration attempt fails. You can run the migration script as many times as necessary without causing any adverse results.

Migration Process Overview

Migrate from WebLogic Collaborate 1.0 or 1.0.1 to WebLogic Integration 2.1 Service Pack 1 (SP1) by completing the following steps:

1. Migrate the B2B repository from WebLogic Collaborate 1.0 or 1.0.1 to WebLogic Integration 2.1 SP1 by following the procedure in “Repository Migration and Workflow Conversion Procedure” on page 3-3. This procedure is represented in the following figure as a gray arrow.

Figure 3-1 Migration from WebLogic Collaborate 1.0 or 1.0.1 to WebLogic Integration 2.1 Service Pack 1



2. Migrate the necessary BPM-related entities, such as the security realm and workflows. See step 9 in “Repository Migration and Workflow Conversion Procedure.”
3. Migrate applications from the C-Enabler API of WebLogic Collaborate Release 1.0 or 1.0.1 to the Messaging API of WebLogic Integration 2.1 by completing the steps in “Supporting the C-Enabler API for WebLogic Collaborate 1.0 or 1.0.1” on page 3-13.

Once WebLogic Collaborate is migrated to WebLogic Integration 2.1, it no longer exists as a single product; rather, it is integrated into WebLogic Integration 2.1 as the component that provides business-to-business integration functionality.

Migrating the Repository and Workflows

WebLogic Integration 2.1 Service Pack 1 provides a utility to help you migrate repository data from a previous release of WebLogic Collaborate to the repository for WebLogic Integration 2.1. The Bulk Migrator utility converts a WebLogic Collaborate 1.0 or 1.0.1 repository data file to a WebLogic Integration 2.1 Service Pack 1 repository data file.

What Is Migrated?

In WebLogic Collaborate Releases 1.0 and 1.0.1, configuration data is provided separately for the following components:

- Repository for the c-hub
- Configuration files for the c-enablers

The following procedure allows you to migrate data only from the c-hub repository. To modify and add data from the c-enabler configuration files and run-time data, you must use the WebLogic Integration B2B Console.

Repository Migration and Workflow Conversion Procedure

1. Use the Bulk Loader from WebLogic Collaborate 1.0 or 1.0.1 to export data from the old repository to a repository data file.

Perform a full (long) repository export. For instructions, see “Working with the Bulk Loader” in the *BEA WebLogic Collaborate C-Hub Administration Guide* in the documentation set for WebLogic Collaborate Release 1.0 or 1.0.1. You can find this document at:

http://e-docs.bea.com/wlcollab/v1_0/chubag/bulkload.htm

or

3 Migrating from WebLogic Collaborate 1.0 or 1.0.1

http://e-docs.bea.com/wlcollab/v1_0_1/chubag/bulkload.htm

As an alternative to running the Bulk Loader, you can use the B2B Console to export the data. For instructions, see “[Configuring B2B Integration](#)” in the BEA WebLogic Integration B2B Console Online Help.

2. Start the WebLogic Server and log on to the WebLogic Integration Studio. Export all workflow templates in XML format. Note the organizations, users, template names, and business operations. This information is required for step 10.
3. Drop the WebLogic Collaborate 1.0-based tables.
4. Install WebLogic Integration 2.1 Service Pack 1 as described in [Installing BEA WebLogic Integration](#).
5. Create the WebLogic Integration 2.1 Service Pack 1 database by running the `createdb` utility in `WLI_HOME/bin`.

Optionally, you can run `createdb` from the Start menu:

- a. Choose Start→Programs→BEA WebLogic E-Business Platform→WebLogic Integration 2.1→Configure.

The BEA WebLogic Integration Database Configuration window is displayed.

- b. Select Create Database and click Next.

The Database Selection window is displayed.

- c. Follow the procedures for creating a database described in “Using the Database Configuration Wizard” in “[Customizing WebLogic Integration](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.

6. Run the Bulk Migrator utility, located in the `WLI_HOME\bin` directory by entering the following command:

```
bulkmigrator.cmd prev_rep_data_file new_rep_data_file
```

In this command:

- `prev_rep_data_file` is the name of the repository data file that you created in step 1.
- `new_rep_data_file` is the name of the new repository data file that is being created.

The new repository data file includes a trading partner element for the old c-hub and for each trading partner that was configured in the old repository.

7. Set the end point values in the new repository data file.

Each trading partner element in the new repository data file includes a value that specifies the trading partner end point, that is, the URL for the trading partner. The Bulk Migrator sets each end point value to `<placeholder>`. You must make sure that each occurrence of `<placeholder>` is set to the correct end point value. If necessary, open the data file in an editor and change the incorrect values to the correct ones.

If you have access to a trading partner's c-enabler configuration file, you can get the URL from this file. The URL is the value of the `ref` attribute for the `enabler-url` XML element. For information about the c-enabler configuration file, see "Configuring C-Enablers" in the *BEA WebLogic Collaborate C-Enabler Administration Guide* in the documentation set for WebLogic Collaborate Release 1.0 or 1.0.1. You can find this document at one of the following URLs:

http://edocs.bea.com/wlcollab/v1_0/cenbag/cfg.htm

or

http://edocs.bea.com/wlcollab/v1_0_1/cenbag/cfg.htm

The following table shows the relationships between the repository data elements for WebLogic Collaborate Release 1.0 or 1.0.1 and the repository data elements for WebLogic Integration 2.1 Service Pack 1.

Table 3-1 Relationships Between Repository Data Elements

WebLogic Collaborate Release 1.0 or 1.0.1	WebLogic Integration 2.1 Service Pack 1
Business protocol	Transport and end point
Business protocol definition	Business protocol definition
C-hub	B2B integration functionality (provided by WebLogic Integration)
Collaborator	Collaboration agreement
Conversation definition	Conversation definition
C-space	Delivery channel

Table 3-1 Relationships Between Repository Data Elements (Continued)

WebLogic Collaborate Release 1.0 or 1.0.1	WebLogic Integration 2.1 Service Pack 1
Extended property set	Extended property set
Logic plug-in	Logic plug-in
Logic plug-in chain	Logic plug-in chain
Role	Role
Subscription	Party
Trading partner	Trading partner
Trading partner protocol	Party identifier
XPath expression	XPath expression

8. Use the Bulk Loader in `WLI_HOME/bin/` to import data from the new repository data file to the repository for WebLogic Integration 2.1 Service Pack 1.

For instructions, see “Working with the Bulk Loader” in the *BEA WebLogic Collaborate C-Hub Administration Guide* in the documentation set for WebLogic Collaborate Release 1.0 or 1.0.1. You can find this document at:

http://e-docs.bea.com/wlcollab/v1_0/chubag/bulkload.htm

or

http://e-docs.bea.com/wlcollab/v1_0_1/chubag/bulkload.htm

As an alternative to running the Bulk Loader, you can use the B2B Console to import the data. For instructions, see “[Configuring B2B Integration](#)” in the BEA WebLogic Integration B2B Console Online Help.

- Note:** After you migrate a WebLogic Collaborate Release 1.0 or 1.0.1 c-hub, the old c-hub is displayed in the B2B Console for WebLogic Integration 2.1 Service Pack 1 as a trading partner named C-Hub.
9. Convert WebLogic Collaborate Release 1.0 workflows. Go to the directory in which the WebLogic Collaborate command scripts are located: `WLI_HOME\bin` (Windows NT) or `WLI_HOME/bin` (UNIX). At the command prompt, enter the command appropriate for your platform:

- Windows

```
PIMigrator.cmd workflow-1.0.xml workflow-2.1.xml
```

- UNIX

```
../PIMigrator.sh workflow-1.0.xml workflow-2.1.xml
```

In the preceding command lines:

- *workflow-1.0.xml* represents the full pathname of the workflow created with WebLogic Collaborate Release 1.0. *workflow-1.0.xml* is generated in step 2.
- *workflow-2.1.xml* represents the full pathname of the workflow converted for use with WebLogic Integration 2.1.

10. Recreate organizations, users, templates, and business operations from the information saved in step 2.
11. Start WebLogic Server and log on to the WebLogic Integration Studio. In the Studio, import the files generated in step 9.

Migrating Applications to the WebLogic Integration 2.1 Messaging API

The WebLogic Integration Messaging API has been updated in Release 2.1 to correctly reflect the new concepts and terminology used for B2B integration functionality in the WebLogic Integration architecture. The C-Enabler API for WebLogic Collaborate Release 1.0 or 1.0.1 is deprecated but still is supported. We recommend that you use the new Messaging API for Java applications. For details about the new messaging API, see [Programming Messaging Applications for B2B Integration](#) and the WebLogic Integration 2.1 Javadoc.

This section describes how to migrate applications from the C-Enabler API for WebLogic Collaborate Release 1.0 or 1.0.1 (which is deprecated but still supported) to the C-Enabler API for WebLogic Integration 2.1 (which is supported). Specifically, this section provides information about the following subjects:

- Introduction to Application Migration

- Importing XOCP Applications
- Updating the Terminology
- Supporting the C-Enabler API for WebLogic Collaborate 1.0 or 1.0.1
- WebLogic Collaborate 1.0 or 1.0.1 C-Enabler API Implementation

Introduction to Application Migration

Note: Any Java messaging application written with the WebLogic Collaborate C-Enabler API that is migrated to WebLogic Integration 2.1 must be run in a separate JVM in nonpersistent mode.

XOCP is the default protocol used by WebLogic Integration for the exchange of business messages. In WebLogic Collaborate Releases 1.0 and 1.0.1, XOCP messaging applications are written using the public XOCP C-Enabler application program interface (API). A message is packaged in an XOCP message envelope and associated with various message headers that specify how to deliver and process the message. In WebLogic Integration 2.1, however, the c-enabler and c-hub entities are integrated and referred to, collectively, as the B2B integration component of WebLogic Integration.

The WebLogic Integration Messaging API now supports C-Enabler functionality and continues to reflect the C-Enabler API run-time terminology so that XOCP applications developed using previous versions of the software will continue to work without additional customization.

The WebLogic Integration Studio gives developers a seamless, reliable method of defining and managing the messaging protocols used during WebLogic Integration business message exchanges. As a result, programming requirements can be reduced considerably. However, the WebLogic Integration Messaging API is also provided so that developers who want to maintain their installations using a Java program as the interface can do so. See [Programming Messaging Applications for B2B Integration](#).

WebLogic Integration uses the XOCP protocol to manage distributed business processes that are being passed back and forth in conversations between trading partners. When appropriately configured, a WebLogic Integration installation verifies business roles and security, providing a central management point for collaborative conversations.

When a business process initiates a specific task (for example, issuing a request for a quote), a central management point is created for collaborative conversations between the trading partners, and business roles and security are verified before a message exchange takes place.

Importing XOCP Applications

The WebLogic Collaborate C-Enabler API does not support the new functionality available in WebLogic Integration 2.1; however, the C-Enabler terminology has been mapped to WebLogic Integration 2.1 functions, and XOCP functions continue to be available from the WebLogic Integration Messaging API. For a complete terminology mapping, see Table 3-2.

The WebLogic Integration Messaging API updates the C-Enabler API primarily in the following areas:

- C-enabler configuration
- Conversation management
- Javadoc

XOCP applications that are being migrated to WebLogic Integration 2.1 must have:

- Database support
- All collaboration agreements defined in the repository

Updating the Terminology

Understanding the new terminology used in WebLogic Integration 2.1 is important when mapping the functionality in XOCP application based on WebLogic Collaborate Release 1.0 or 1.0.1 for migration purposes. The configuration characteristics of your

3 Migrating from WebLogic Collaborate 1.0 or 1.0.1

WebLogic Integration installation determine whether functionality previously associated with the c-enabler or the c-hub is invoked. Table 3-2 summarizes the mapping of pre-2.1 terms to Release 2.1.

Table 3-2 Terminology Mappings from WebLogic Collaborate 1.0 and 1.0.1 to WebLogic Integration 2.1

WebLogic Collaborate 1.0 or 1.0.1 Term	WebLogic Integration 2.1 Term	Definition
Attachment	Attachment	NonXML payload part of a business message; for example, a GIF file.
Business document	Business document	XML-based payload part of a business message.
Business message	Business message	A multipart MIME message, consisting of one or more payload parts. A part can be an XML document (a business document) or nonXML file (an attachment).
Business protocol definition	Business protocol definition	The definition of a business transaction protocol, for example, Name=XOCP.
Collaborator	Trading partner session	In WebLogic Collaborate Releases 1.0 and 1.0.1, a collaborator is a trading partner that participates in a single c-space (delivery channel). In WebLogic Integration 2.1, a trading partner session is created to support communication with each trading partner, and its scope is bounded by a delivery channel.
Conversation	Conversation	A series of message exchanges between trading partners.
Conversation definition	Conversation definition	A set of roles and document definitions, that are grouped for a particular purpose, such as a purchase order.

Table 3-2 Terminology Mappings from WebLogic Collaborate 1.0 and 1.0.1 to WebLogic Integration 2.1 (Continued)

WebLogic Collaborate 1.0 or 1.0.1 Term	WebLogic Integration 2.1 Term	Definition
Conversation type	Not applicable	There is no corresponding term for conversation type in WebLogic Integration 2.1. In WebLogic Collaborate Releases 1.0 and 1.0.1, a conversation type defined a conversation for a trading partner. It consisted of the following properties: <ul style="list-style-type: none"> ■ Conversation name (for example, purchase order) ■ Version ■ Role of a trading partner in the specified conversation (for example, buyer or seller)
C-Space	Delivery channel	A grouping of a trading partner, a subscription, and a business protocol.
C-enabler node	Spoke	A trading partner that communicates with others through a hub (intermediary).
C-Enabler API	WebLogic Integration Messaging API	In WebLogic Collaborate Releases 1.0 and 1.0.1, the C-Enabler API is the interface between an XOCP application and WebLogic Collaborate at run time. This API is used, for example, to start c-enabler sessions, create and participate in conversations, and send and receive business messages. XOCP applications based on the C-Enabler API can be migrated to use the WebLogic Integration Messaging API.

Table 3-2 Terminology Mappings from WebLogic Collaborate 1.0 and 1.0.1 to WebLogic Integration 2.1 (Continued)

WebLogic Collaborate 1.0 or 1.0.1 Term	WebLogic Integration 2.1 Term	Definition
C-enabler Session	Trading partner session	In WebLogic Collaborate Releases 1.0 and 1.0.1, a c-enabler session is a logical session conducted by a c-enabler node and one c-hub for a particular c-space. In WebLogic Integration 2.1, a trading partner session is conducted in a delivery channel.
C-hub	Intermediary	A WebLogic Server running a set of shared services, such as: <ul style="list-style-type: none"> ■ Repository ■ Administration ■ Conversation coordination, message routing, and mediation
Message	Message	Data, in multipart MIME format, used for communication.
Role	Role	A part played by a collaborator in a conversation. The part is defined by the set of documents that can be sent and received by the collaborator performing the part. A minimum of two roles, such as buyer and seller, are associated with each conversation.
Send(Publish)-side filter	Send(Publish)-side filter	Filter (X-Path) specified by an application to determine the list of trading partners to receive a document. For example: <code>//trading partner[@name="Bea "</code>
Trading partner	Trading partner	A representation of an entity, such as a company, that participates in one or more delivery channels.

Supporting the C-Enabler API for WebLogic Collaborate 1.0 or 1.0.1

In WebLogic Integration 2.1, the configuration of a trading partner is defined and entered in the system using the B2B Console. The trading partner retrieves a collaboration agreement from the WebLogic Integration repository.

As a candidate for migration to WebLogic Integration 2.1, a C-enabler application based on WebLogic Collaborate 1.0 or 1.0.1 must have existing database support, and all of its trading partner collaboration agreements must be predefined in the repository. To collaborate in business transactions using the XOCP protocol, the application must be aligned with the WebLogic Integration Messaging API. To make an existing XOCP application ready for migration to the WebLogic Integration Messaging API, changes are required in the configuration of the C-enabler and the repository.

C-Enabler Configuration

In WebLogic Collaborate Releases 1.0 and 1.0.1, the c-enabler configuration is retrieved from an XML file that defines the following variables:

- session name
- c-space name—Name of the intermediary delivery channel in the collaboration agreement.
- hub-url—One or more endpoint URLs of the intermediary delivery channel
- enabler-url—One or more endpoint URLs of the spoke delivery channel
- security-info—Security information in the document exchange elements of the delivery channels of both the spoke and the intermediary
 - trading partner name
 - certificate
 - role

Then the c-enabler creates (or participates in) and manages conversations with other trading partners.

In WebLogic Integration 2.1, the identification and configuration of a trading partner is defined and entered in the system using the B2B Console. The trading partner retrieves its collaboration agreement from the WebLogic Integration repository.

Repository

Existing XOCP applications can be run with the WebLogic Integration 2.1 repository with only one change: WebLogic Integration must have a repository, and all the trading partner collaboration agreements used by the XOCP applications must be defined in the repository. To migrate a data repository for WebLogic Collaborate Release 1.0 and 1.0.1 to WebLogic Integration 2.1, use the Bulk Migrator utility described in “Migrating the Repository and Workflows” on page 3-3.

WebLogic Collaborate 1.0 or 1.0.1 C-Enabler API Implementation

This section lists the public API packages, interfaces, and classes provided by WebLogic Collaborate Release 1.0 or 1.0.1 that must be reintegrated in the WebLogic Integration 2.1 run-time implementation in order to migrate XOCP applications.

Required Packages

- `com.bea.b2bcom.bea.b2b.enabler`
- `com.bea.b2b.enabler.xocp`
- `com.bea.b2b.protocol`
- `com.bea.b2b.protocol.conversation`
- `com.bea.b2b.protocol.messaging`
- `com.bea.b2b.protocol.xocp.conversation.local`
- `com.bea.b2b.protocol.xocp.messaging`

Required Interfaces

- `com.bea.b2b.EnablerSession`

- `com.bea.b2b.enabler.xocp.XOCPEablerSession`
- `com.bea.b2b.protocol.PlugIn`
- `com.bea.b2b.protocol.conversation.State`
- `com.bea.b2b.protocol.messaging.Message`
- `com.bea.b2b.protocol.messaging.PayloadPart`
- `com.bea.b2b.protocol.xocp.conversation.local.Conversation`
- `com.bea.b2b.protocol.xocp.conversation.local.ConversationHandler`

Required Classes

- `com.bea.b2b.enabler.Enabler`
- `com.bea.b2b.protocol.conversation.ConversationType`
- `com.bea.b2b.protocol.messaging.Attachment`
- `com.bea.b2b.protocol.messaging.BusinessDocument`
- `com.bea.b2b.protocol.messaging.DeliveryStatus`
- `com.bea.b2b.protocol.messaging.MessageBase`
- `com.bea.b2b.protocol.messaging.MessageEnvelope`
- `com.bea.b2b.protocol.messaging.MessageToken`
- `com.bea.b2b.protocol.messaging.Miniparser`
- `com.bea.b2b.protocol.xocp.messaging.Ping`
- `com.bea.b2b.protocol.xocp.messaging.QualityOfService`
- `com.bea.b2b.protocol.xocp.messaging.XOCPMMessage`
- `com.bea.b2b.protocol.xocp.messaging.XOCPMMessageToken`

4 Migrating from WebLogic Integration 2.0 to WebLogic Integration 2.1 Service Pack 1

This section provides two procedures for migrating from BEA WebLogic Integration 2.0 to BEA WebLogic Integration 2.1 Service Pack 1 (SP1). This section includes the following topics:

- Migrating from WebLogic Integration 2.0 with No Service Pack or Service Pack 1 to WebLogic Integration 2.1 Service Pack 1
- Migrating from WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to WebLogic Integration 2.1 Service Pack 1

Migrating from WebLogic Integration 2.0 with No Service Pack or Service Pack 1 to WebLogic Integration 2.1 Service Pack 1

To migrate WebLogic Integration 2.0 with no service pack or with Service Pack 1 (SP1) to WebLogic Integration 2.1 SP1, complete the following procedure:

1. Migrate from WebLogic Integration 2.0 with no service pack or with SP1 to WebLogic Integration 2.0 Service Pack 4 (SP4) by following the directions in “Migration Issues” in the Release 2.0 SP4 edition of the *BEA WebLogic Integration Release Notes*. This document is available at the following URL:

http://websupport.bea.com/custsupp/docs/wlintegration/v2_0sp4/relnotes/appa.htm

Access to this URL requires a BEA WebSUPPORT login and password. BEA WebSUPPORT is one of the many benefits provided to contract customers of BEA Customer Support. For more information about BEA WebSUPPORT, go to the following URL:

<http://support.bea.com/welcome.jsp>

2. Migrate from WebLogic Integration 2.0 SP4 to WebLogic Integration 2.1 SP1 by following the directions in “Migrating from WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to WebLogic Integration 2.1 Service Pack 1” on page 4-3.

Migrating from WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to WebLogic Integration 2.1 Service Pack 1

This section provides a procedure for migrating from BEA WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to BEA WebLogic Integration 2.1 Service Pack 1. It contains information about the following subjects:

- Migration Process Overview
- Before You Begin
- Database Migration

Migration Process Overview

In addition to providing a direct migration procedure from WebLogic Integration 2.0 Service Pack 2, 3, or 4 to WebLogic Integration 2.0 Service Pack 1, the procedures provided in “Database Migration” on page 4-5 can also be used to complete the overall migration processes described in the following chapters and sections:

- Chapter 2, “Migrating from WebLogic Process Integrator 1.2 or 1.2.1.”
- Chapter 3, “Migrating from WebLogic Collaborate 1.0 or 1.0.1.”
- “Migrating from WebLogic Integration 2.0 with No Service Pack or Service Pack 1 to WebLogic Integration 2.1 Service Pack 1.”

For example, when migrating from WebLogic Integration 2.0 with no service pack or with Service Pack 1, you must first migrate to WebLogic Integration 2.0 Service Pack 4 and then migrate to WebLogic Integration 2.1 Service Pack 1.

Before You Begin

Before following the procedures provided in “Migrating with the Database Configuration Wizard” on page 6-1 or “Migrating with the Command-Line Migration Script” on page 6-7, you must consider the following:

- Domain Migration
- Remote Database Server
- Security File Realms

Domain Migration

The procedures provided in Chapter 6, “Migration Utilities,” run only migration utilities that migrate the WebLogic Integration repository information stored in database tables. These utilities do not migrate WebLogic Server domain information for your application.

To migrate the WebLogic Server domain information for your application, complete the following procedure:

1. Migrate the WebLogic Integration 2.0 repository information to one of the preconfigured WebLogic Integration 2.1 domains, as described in “Database Migration” on page 4-5. For more information about domains, see “WebLogic Server Domains” in “[Getting Started](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.
2. Copy any application-specific entries from the `config.xml` file for your existing WebLogic Integration 2.0 domain to the `config.xml` file for your existing WebLogic Integration 2.1 domain.

WebLogic Integration 2.1 is delivered with four preconfigured domains: `bpmdomain`, `eaideomain`, `samples`, and `wlideomain`. The `config.xml` files for these domains are located in the `WLI_HOME/config/DOMAIN_NAME` directory, where `DOMAIN_NAME` is `bpmdomain`, `eaideomain`, `samples`, or `wlideomain`. For example, the `config.xml` file for the `wlideomain` is located in the `WLI_HOME/config/wlideomain` directory.

Remote Database Server

If the machine on which WebLogic Integration 2.1 Service Pack 1 is installed and on which you plan to run the migration utility is different from the machine that hosts the database server with the WebLogic Integration repository, you must meet the following prerequisites before running the migration utility:

- A client installation of the database product must be installed on the WebLogic Integration machine.
- The database directory that contains the command-line executables for the database must be included in the `PATH` environment variable. For example, for the Oracle database on a UNIX platform, `$ORACLE_HOME/bin` must be included in the `PATH`.

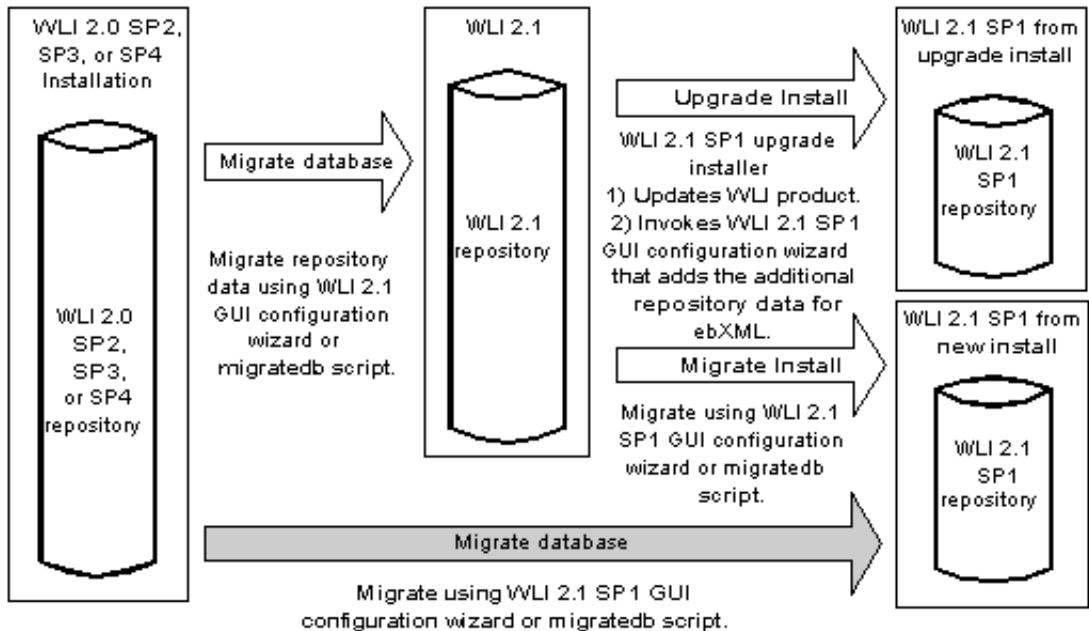
Security File Realms

If you are using the file realm for Security, you must complete one more task before starting a migration procedure (either “Migrating with the Database Configuration Wizard” on page 6-1 or “Migrating with the Command-Line Migration Script” on page 6-7). Before beginning either procedure, you need to merge the WebLogic Integration 2.0 `filerealm.properties` file with the `filerealm.properties` file for WebLogic Integration 2.1 in `WLI_HOME/config/DOMAIN_NAME`.

Database Migration

The migration utilities provided with WebLogic Integration 2.0 Service Pack 1 provide a direct migration path from WebLogic Integration Service Pack 2, 3, or 4 to WebLogic Integration 2.0 Service Pack 1. This direct migration path is represented by the gray arrow labeled `Migrate database` in the following figure.

Figure 4-1 Direct Migration Path from WebLogic Integration 2.0 Service Pack 2, 3, or 4 to WebLogic Integration 2.1 Service Pack 1



If you are currently running WebLogic Integration 2.0 with Service Pack 2, 3, or 4, you can migrate to WebLogic Integration 2.1 by following the procedure in either “Migrating with the Database Configuration Wizard” on page 6-1 or “Migrating with the Command-Line Migration Script” on page 6-7.

5 Migrating WebLogic Integration 2.1 to WebLogic Integration 2.1 Service Pack 1

This section describes the procedure for migrating from BEA WebLogic Integration 2.1 to BEA WebLogic Integration 2.1 Service Pack 1.

It includes the following topics:

- Before You Begin
- Domain Migration
- Database Migration

Before You Begin

Before you attempt to migrate to WebLogic Integration 2.1 Service Pack 1, we strongly recommend that you back up your entire database and export all your workflows. This preparation will enable you to fix your environment and start the migration again if your first migration attempt fails. You can run the migration script as many times as necessary without any adverse results.

Domain Migration

The procedures provided in “Database Migration” on page 5-3 run only migration utilities that migrate the WebLogic Integration repository information stored in database tables. These utilities do not migrate WebLogic Server domain information for your application. For example, if you create a domain for your WebLogic Integration 2.1 application and then run the migration procedure on the database for that domain, you must also update the `config.xml` file for the application domain by adding the additional entries needed for WebLogic Integration 2.1 Service Pack 1 (SP1). The procedure for adding those additional entities is provided in this section.

If, however, you are using one of the preconfigured domains installed with WebLogic Integration 2.1 and you use the upgrade installer to upgrade to WebLogic Integration 2.1 SP1, you do not need to follow the procedure in this section because the upgrade installer automatically updates the `config.xml` files for the four preconfigured domains with the additional entities needed for WebLogic Integration 2.1 SP1.

To migrate the WebLogic Server domain information for your application, complete the following procedure:

1. In a text editor, open the `config.xml` file for your WebLogic Integration 2.1 application domain.
2. Edit the `config.xml` file.

Note: Any of the `config.xml` files for the four preconfigured domains can be used as a guide to the changes you make in this step. Each `config.xml` file is located in the appropriate `WLI_HOME/config/DOMAIN_NAME` directory, where `WLI_HOME` is the directory in which you installed WebLogic Integration and `DOMAIN_NAME` is `bpmdomain`, `eidomain`, `samples`, or `wlidomain`. For example, the `config.xml` file for the `wlidomain` is located in the `WLI_HOME/config/wlidomain` directory.

Make the following changes:

- a. Add a deployment entry for the ebXML plug-in jar file in two steps: (1) add the two lines highlighted (in bold) in Listing 5-1, and (2) increment the values of `DeploymentOrder` for all the remaining deployment entries, as shown in Listing 5-1.

b. Change the name of the data integration plug-in from `XTPlugin` to `com.bea.wlxt.WLXTPlugin`, as shown in the last deployment entry of Listing 5-1.

Listing 5-1 Deployment Entries in the `config.xml` File After Editing

```
<EJBComponent Name="wlaipugin-ejb.jar" Targets="myserver"
  URI="wlaipugin-ejb.jar" DeploymentOrder="9"/>
<EJBComponent Name="ebxml-bpm-plugin.jar" Targets="myserver"
  URI="ebxml-bpm-plugin.jar" DeploymentOrder="10"/>
<WebAppComponent Name="b2bconsole" ServletReloadCheckSecs="1"
  Targets="myserver" URI="b2bconsole.war" DeploymentOrder="11"/>
<WebAppComponent Name="wlai" ServletReloadCheckSecs="1"
  Targets="myserver" URI="wlai.war" DeploymentOrder="12"/>
<WebAppComponent Name="WLAIPugin" Targets="myserver"
  URI="wlai-plugin.war" DeploymentOrder="13"/>
<WebAppComponent Name="com.bea.wlxt.WLXTPlugin" Targets="myserver"
  URI="wlxtpi.war" DeploymentOrder="14"/>
```

c. Add a new JMS queue by inserting the line that is highlighted (in bold) in Listing 5-2.

Listing 5-2 Addition of JMS Queue in the `config.xml` File

```
<JMSTopic JNDIName="com.bea.wlpi.NotifyTopic" Name="wlpiNotify"/>
<JMSQueue JNDIName="com.bea.b2b.OutboundQueue" Name="B2bOutboundQueue"/>
</JMSServer>
```

Database Migration

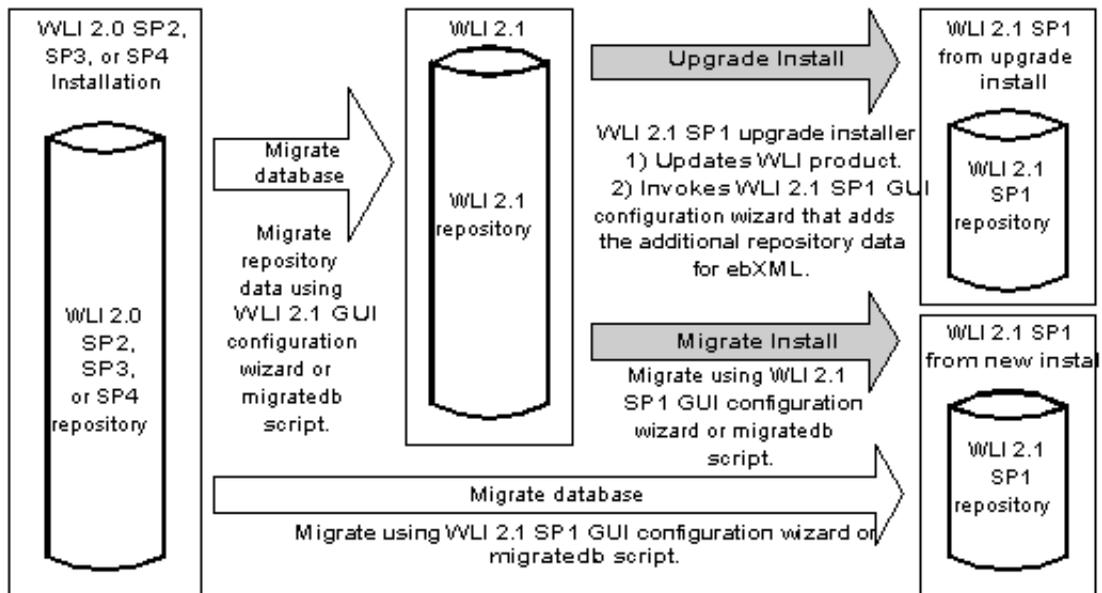
WebLogic Integration 2.1 Service Pack 1 (SP1) requires database entities not required for WebLogic Integration 2.1.

To convert your installation of WebLogic Integration 2.1 to WebLogic Integration 2.1 SP1, you must run one of the following procedures:

- Upgrade Install** — Use the update installer to convert your installation of WebLogic Integration 2.1 to WebLogic Integration 2.1 SP1. The update installer optionally invokes a migration utility that updates the existing WebLogic Integration 2.1 repository database tables with the additional database entries needed for WebLogic Integration 2.1 SP1. This upgrade procedure is represented by the gray arrow labeled Upgrade Install in Figure 5-1.
- New Install** — A new version of WebLogic Integration 2.1 SP 1 is installed. One of the migration utilities is used to update the existing WebLogic Integration 2.1 repository database tables with the additional database entries needed for WebLogic Integration 2.1 SP1. This upgrade procedure is represented by the gray arrow labeled Migrate Install in Figure 5-1.

Caution: Both of these migration procedures update a single repository. The migration of repository data from an existing database instance to a new instance is not supported.

Figure 5-1 WebLogic Integration 2.1 to WebLogic Integration 2.1 Service Pack 1 Database Migration



Upgrade Install

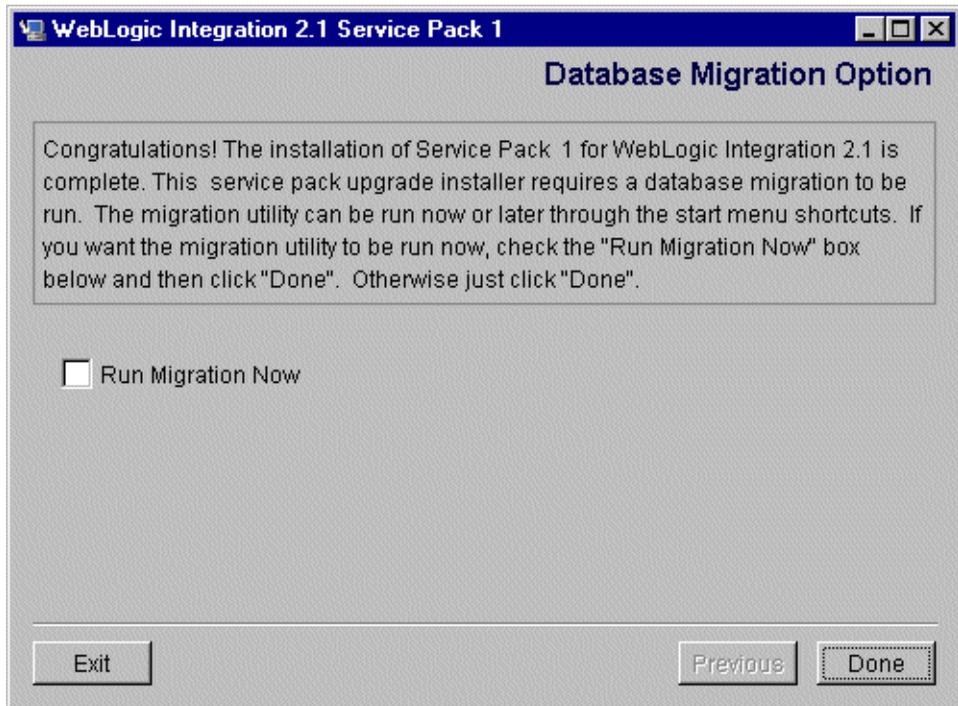
This section provides a procedure for using the service pack upgrade installer to convert your installation of WebLogic Integration 2.1 to WebLogic Integration 2.1 Service Pack 1 (SP1). The conversion includes the upgrade of the repository data so it becomes compatible with WebLogic Integration 2.1 SP1.

To upgrade and migrate the repository data:

1. Run the Service Pack Upgrade Installer. For more information see [“Installing and Uninstalling Service Pack Upgrades”](#) in *Installing BEA WebLogic Integration*.
2. Migrate the repository data during the service pack upgrade, or run one of the migration utilities after completing the service pack upgrade.

The service pack upgrade installer gives you the option of invoking the WebLogic Integration database configuration wizard directly from the service pack upgrade installer, as shown in Figure 5-2.

Figure 5-2 Invoking the Database Configuration Wizard to Migrate with the Service Pack Upgrade Installer



Migrate your repository data by following one of these procedures:

- Select the Run Migration Now check box to invoke the database configuration wizard from the service pack upgrade installer.
- After completing the service pack upgrade procedure invoke one of the following migration utilities: the database configuration wizard or the migration command script.

If you select the Run Migration Now check box, follow the procedure provided in “Migrating with the Database Configuration Wizard” on page 6-1, starting with step 2.

If you do not select the Run Migration Now check box, complete the service pack upgrade, and then run a migration utility. Run either the WebLogic Integration 2.1 database configuration wizard or the migration script to update

the existing WebLogic Integration 2.1 repository database tables with the additional database entries needed for WebLogic Integration 2.1 SP1. Follow one of the migration procedures provided in Chapter 6, “Migration Utilities.”

Caution: You must complete the database migration before you can start to use the upgraded WebLogic Integration 2.1 SP1 installation.

New Installation

This section provides a procedure for updating the repository data of an existing WebLogic Integration 2.1 installation to make it compatible with a new installation of WebLogic Integration 2.1 Service Pack 1 (SP1).

To update the repository data for a new installation of WebLogic Integration 2.1 SP1:

1. Install WebLogic Integration 2.1 SP1. For more information, see [Installing BEA WebLogic Integration](#).

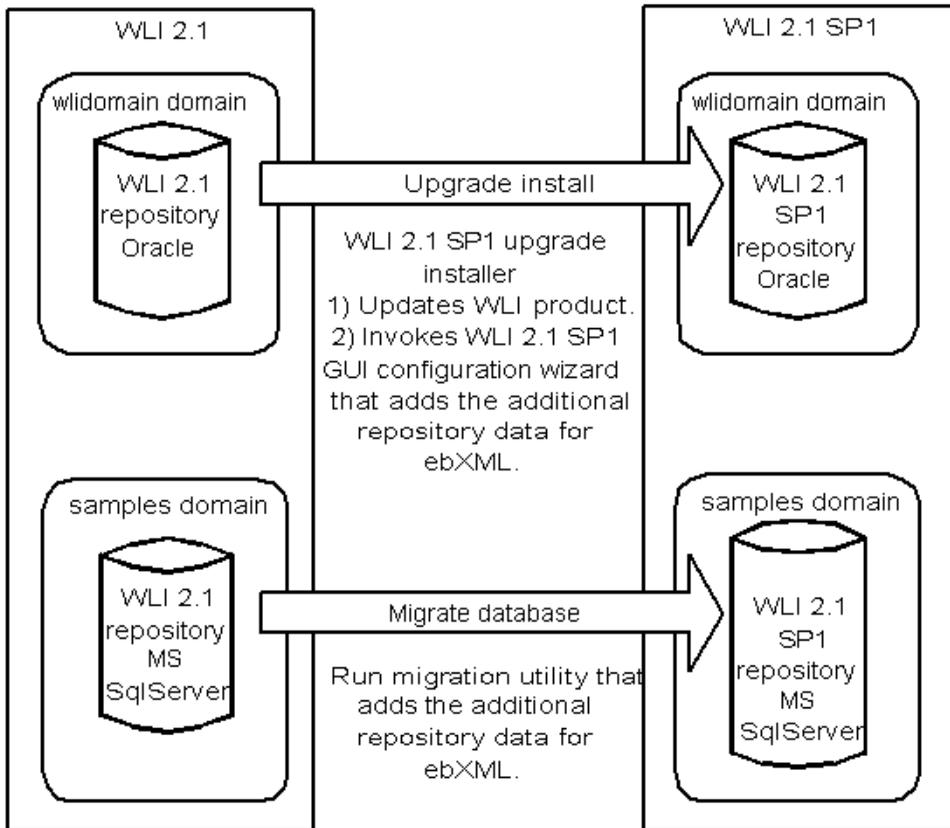
Note: Do not create a new WebLogic Integration repository or database. Do not run the `RunSamples` script.

2. Run a migration utility (either the WebLogic Integration 2.1 database configuration wizard or the migration script) to update the existing WebLogic Integration 2.1 repository database tables with the additional database entries needed for WebLogic Integration 2.1 SP1. Follow one of the migration procedures provided in Chapter 6, “Migration Utilities.”

Migrating Two Repositories

This section provides a procedure for updating the repository data of an existing WebLogic Integration 2.1 installation with two repositories to be compatible with a new installation of WebLogic Integration 2.1 Service Pack 1 (SP1). For example, you may have two repositories, one for deploying your WebLogic Integration application in a production environment and one for developing your WebLogic Integration application. For your production environment, the repository resides in an Oracle database of the `wlidomain` domain and for the development environment, the repository resides in a Microsoft SQL Server database of the `samples` domain. Figure 5-3 illustrates this scenario.

Figure 5-3 Example Database Configuration



To migrate the two database instances for the example configuration shown in Figure 5-3, complete the following procedure:

1. Migrate the Oracle database using the upgrade installer to convert your installation of WebLogic Integration 2.1 to WebLogic Integration 2.1 SP1. Follow the procedure provided in “Upgrade Install” on page 5-5.
2. Migrate the Microsoft SQL Server database by running a migration utility (either the WebLogic Integration 2.1 database configuration wizard or the migration script). The database entries needed for WebLogic Integration 2.1 SP1 are added

to the existing WebLogic Integration 2.1 repository database tables. To perform this step, follow one of the migration procedures provided in Chapter 6, “Migration Utilities.”

Warning: Make sure you switch the default domain before running a migration utility. In the example configuration shown in Figure 5-3, the default domain would change from the `wl1domain` to `samples` domain. The migration procedures in Chapter 6, “Migration Utilities,” provide steps for switching the default domain.

6 Migration Utilities

This section provides procedures for using the migration utilities. It includes the following topics:

- Migrating with the Database Configuration Wizard
- Migrating with the Command-Line Migration Script

Migrating with the Database Configuration Wizard

To migrate to WebLogic Integration 2.1 Service Pack 1 (SP1) using the WebLogic Integration database configuration wizard in graphical mode, complete the following procedure:

1. Start the WebLogic Integration database configuration wizard by completing the procedure appropriate for your platform:
 - Windows:
 - a. Select the domain that contains the database to be migrated. For more information about the available domains, see “WebLogic Server Domains” in *“Getting Started”* in *Starting, Stopping, and Customizing BEA WebLogic Integration*.
 - b. Start the database configuration wizard for the appropriate domain.

For this domain . . .	Choose . . .
wlidomain	Start→BEA WebLogic E-Business Platform→WebLogic Integration 2.1→Configure

For this domain . . .	Choose . . .
samples	Start→BEA WebLogic E-Business Platform→WebLogic Integration 2.1→Samples→Configure
eaidomain	Start→BEA WebLogic E-Business Platform→WebLogic Integration 2.1→EAI Domain→Configure
bpmdomain	Start→BEA WebLogic E-Business Platform→WebLogic Integration 2.1→BPM Domain→Configure

The Choose Configuration Option dialog box is displayed.

- UNIX:

- a. Execute the following commands:

```
cd WLI_HOME/bin  
wliconfig
```

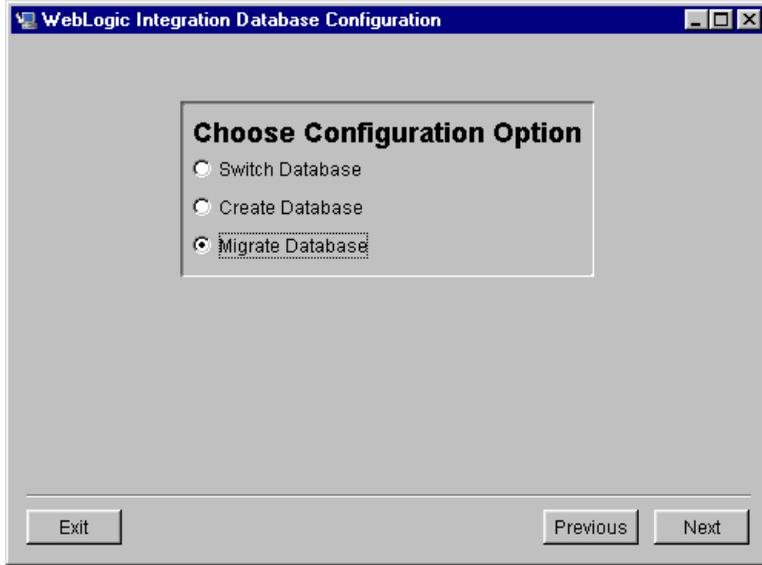
The Choose BEA Home Directory dialog box is displayed.

- b. Select an existing BEA Home directory, and then click Next.

The Choose Domain to Configure dialog box is displayed.

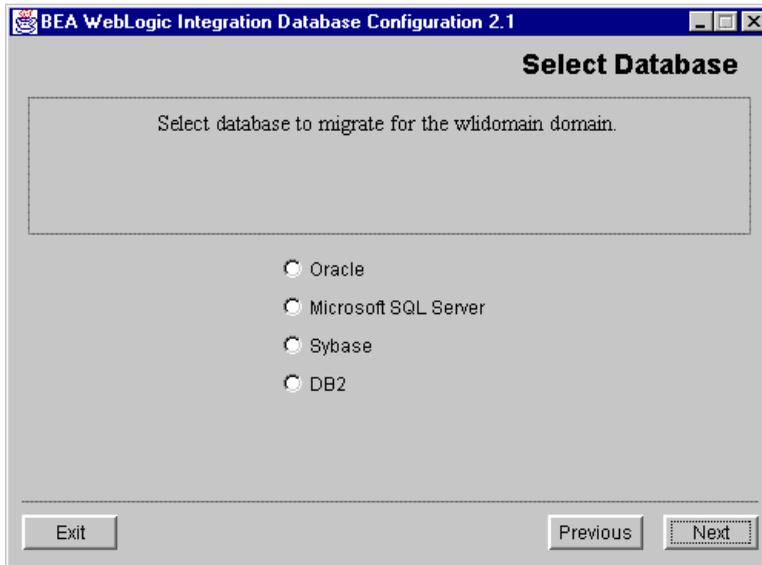
- c. Select a domain, and then click Next. For more information about the available domains, see “WebLogic Server Domains“ in “[Getting Started](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.

The Choose Configuration Option dialog box is displayed.



2. Select Migrate Database and click Next.

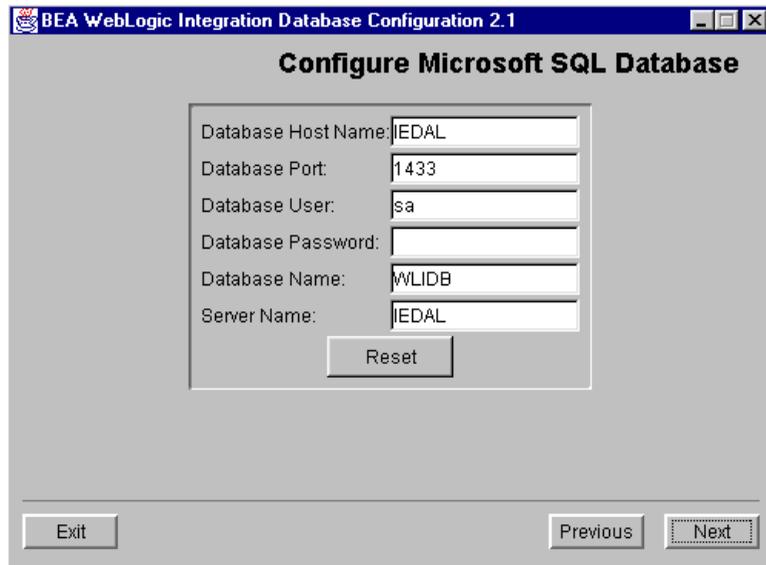
The Select Database dialog box is displayed.



3. Select a database type (Oracle, Microsoft SQL Server, or Sybase) and select Next.

Note: Migration is not available for Cloudscape or DB2.

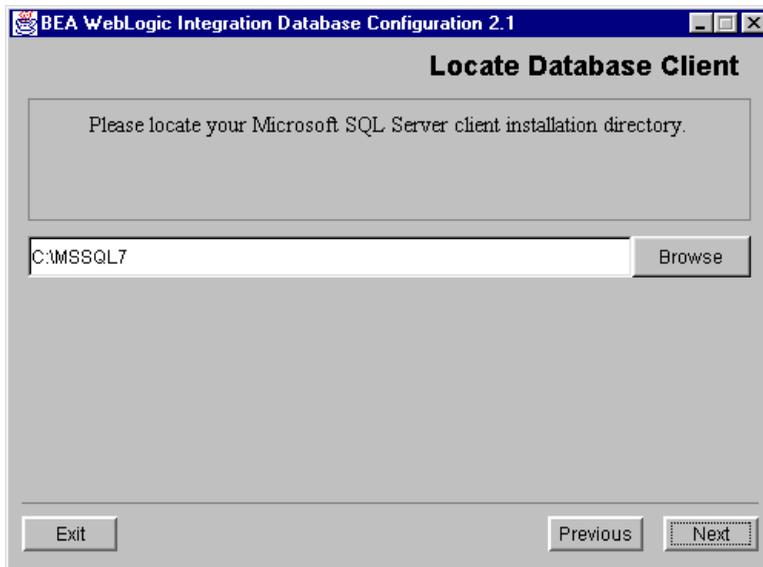
The Configure *database_type* Database dialog box is displayed. The name of the database type you selected replaces *database_type* in the window title. In this example Microsoft SQL is selected, so the Configure Microsoft SQL Database dialog box is displayed.



The screenshot shows a dialog box titled "BEA WebLogic Integration Database Configuration 2.1" with a sub-title "Configure Microsoft SQL Database". The dialog box contains the following fields and buttons:

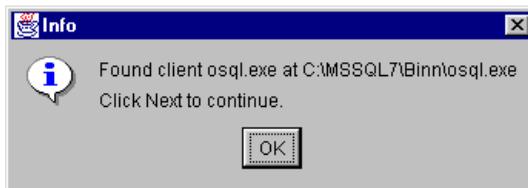
- Database Host Name: IEDAL
- Database Port: 1433
- Database User: sa
- Database Password: (empty)
- Database Name: WLIDB
- Server Name: IEDAL
- Reset button
- Exit button
- Previous button
- Next button

4. You are prompted to provide database-specific information. As shown in the preceding dialog box, for example, you may be prompted to specify a host, port, user, password, database name, and server. For more information about entering values in the fields in this dialog box, see Table 2-2 in “Preconfigured Domains and Database Selection“ in “[Installing WebLogic Integration Using Graphics Mode](#)” in *Installing BEA WebLogic Integration*. Enter values in the fields displayed and select Next.
5. The Locate Database Client dialog box is displayed.



6. If necessary, select Browse and locate the directory containing the selected database client installation. Select Next.

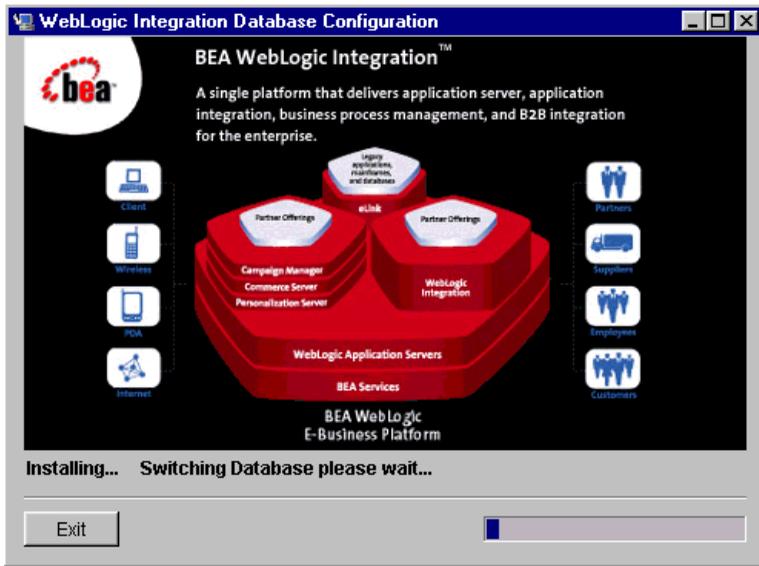
When the database is found a confirmation message is displayed.



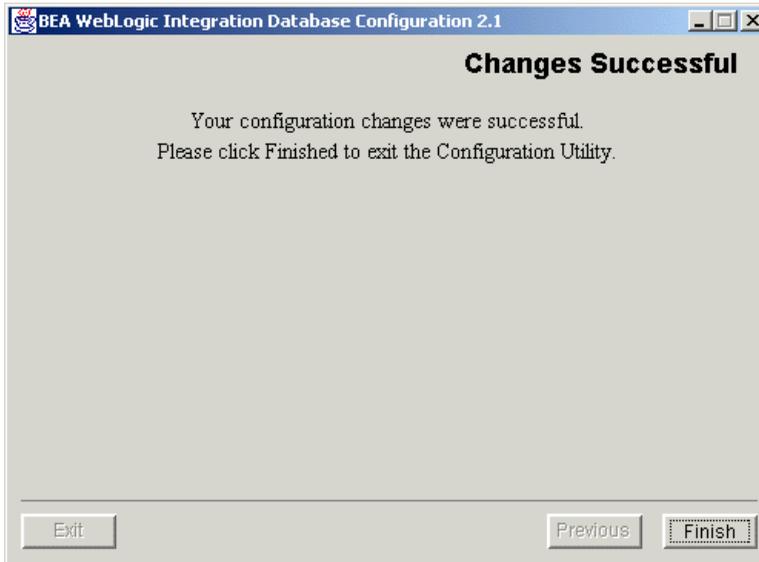
Note: If it is not necessary for you to locate your client installation directory, then this confirmation message is displayed immediately after step 6, and the Locate Database Client dialog box is not displayed.

7. Click OK. You are returned to the Locate Database Client window. Click Next.

The installation process begins and a BEA WebLogic Integration 2.1 splash screen is displayed.



The splash screen remains visible until the progress meter in the lower right corner is completely filled. Then the Changes Successful message window is displayed.



-
8. Click Finish to end the migration process.

Your database migration to WebLogic Integration 2.1 SP1 is now complete.

Migrating with the Command-Line Migration Script

This section provides two operating system-specific procedures. Complete the procedure appropriate for your platform:

- Windows
- UNIX

Windows

To run the migration script manually on a Windows system:

1. Go to the `bin` directory in your WebLogic Integration home directory. For example:

```
cd c:\bea\wlintegration2.1\bin
```

2. Select one of the preconfigured WebLogic Integration 2.1 domains. For more information, see “WebLogic Server Domains“ in “[Getting Started](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.

3. Run the `setdomain` script on the domain selected in the previous step. The path for the domain depends on your installation. For example:

```
setdomain c:\bea\wlintegration2.1\config\wlidomain
```

4. Run the migration script to migrate the repository data for WebLogic Integration 2.0 with Service Pack 2, 3, or 4 to the WebLogic Integration 2.1 SP1 domain:

```
migratedb
```

UNIX

To run the migration script manually on a UNIX system:

1. Go to the `bin` directory in the WebLogic Integration home directory. For example:

```
cd /home/me/bea/wlintegration2.1/bin
```

2. Select one of the preconfigured WebLogic Integration 2.1 domains. For more information, see “WebLogic Server Domains“ in “[Getting Started](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.
3. Run the `setdomain` script on the domain selected in the previous step. The path for the domain depends on your installation. For example:

```
setdomain /home/me/boa/wlintegration2.1/config/wlidomain
```

4. Run the migration script to migrate the repository data to the WebLogic Integration 2.1 Service Pack 1 domain:

```
migratedb
```