



# BEA WebLogic Integration™

## Migrating to BEA WebLogic Integration Release 2.1

## Copyright

Copyright © 2001 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

<b>Part Number</b>	<b>Date</b>	<b>Software Version</b>
N/A	Release: October 2001 Revised: November 30, 2001	2.1

---

# Contents

## About This Document

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

## 1. Introduction to Migration

Migration Overview .....	1-1
Why Migrate? .....	1-2
Migration Scenarios .....	1-2

## 2. Migrating from WebLogic Process Integrator 1.2/1.2.1

Migration Process .....	2-1
Before You Begin .....	2-2
Migrating Your Database .....	2-2
Migrating the Database Schema .....	2-3
Migrating the Security Realm .....	2-4
Select a Migration Procedure .....	2-4
Migrate from RDBMS Realm (WebLogic Process Integrator 1.2 or 1.2.1) .....	2-5
Migrate from File Realm (WebLogic Process Integrator 1.2.1 Only) .....	2-7
Assigning Permissions .....	2-8

---

### 3. Migrating from WebLogic Collaborate 1.0/1.0.1

Migration Process .....	3-1
Before You Begin .....	3-2
Migrating the Repository and Workflows .....	3-2
What Is Migrated? .....	3-2
Repository Migration Procedure .....	3-3
Migrating Applications to the WebLogic Integration 2.1 Messaging API.....	3-7
Introduction .....	3-7
Importing XOCP Applications .....	3-8
Updating the Terminology .....	3-9
Supporting the WebLogic Collaborate 1.0/1.0.1 C-Enabler API.....	3-12
C-Enabler Configuration .....	3-13
Repository .....	3-13
WebLogic Collaborate 1.0/1.0.1 C-Enabler API Implementation .....	3-14
Packages .....	3-14
Interfaces .....	3-14
Classes .....	3-14

### 4. Migrating from WebLogic Integration 2.0 (No SP or SP 1) to WebLogic Integration 2.1

### 5. Migrating from WebLogic Integration 2.0 (SP 2, 3 and 4) to WebLogic Integration 2.1

Overview .....	4-2
Before You Begin .....	4-2
Domain Migration .....	4-3
Remote Database Server .....	4-3
Security File Realms .....	4-4
Migrating Using the WebLogic Integration	
Database Configuration Wizard in Graphical Mode .....	4-4
Running the Migration Script Manually.....	4-10
Windows.....	4-10
UNIX.....	4-11

---

# About This Document

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

- Chapter 1, “Introduction to Migration,” provides background on migrating WebLogic Process Integrator 1.2/1.2.1 and WebLogic Collaborate 1.0/1.0.1 to BEA WebLogic Integration 2.1.
- Chapter 2, “Migrating from WebLogic Process Integrator 1.2/1.2.1,” describes the procedures for migrating WebLogic Process Integrator 1.2/1.2.1 to BEA WebLogic Integration 2.1.
- Chapter 3, “Migrating from WebLogic Collaborate 1.0/1.0.1,” describes the procedures for migrating WebLogic Collaborate 1.0/1.0.1 to BEA WebLogic Integration 2.1.
- Chapter 4, “Migrating from WebLogic Integration 2.0 (No SP or SP 1) to WebLogic Integration 2.1,” describes the procedures for migrating WebLogic Integration 2.0 with no service pack or with service pack 1 to WebLogic Integration 2.1.
- Chapter 5, “Migrating from WebLogic Integration 2.0 (SP 2, 3 and 4) to WebLogic Integration 2.1,” describes the procedures for migrating service packs 2, 3, and 4 of WebLogic Integration 2.0 to WebLogic Integration 2.1.

---

# What You Need to Know

*Migrating to BEA WebLogic Integration Release 2.1* is designed primarily for use by WebLogic Integration users who want to migrate previous versions of the product—primarily WebLogic Process Integrator 1.2/1.2.1, WebLogic Collaborate 1.0/1.0.1 to BEA WebLogic Integration 2.1 and WebLogic Integration 2.0—to 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](mailto: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](http://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.

<b>Convention</b>	<b>Item</b>
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 &lt;iostream.h&gt; void main ( ) the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float</pre>
<b>monospace boldface text</b>	Identifies significant words in code. <i>Example:</i> <pre>void <b>commit</b> ( )</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"> <li>■ That an argument can be repeated several times in a command line</li> <li>■ That the statement omits additional optional arguments</li> <li>■ That you can enter additional parameters, values, or other information</li> </ul> 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 section contains information on the following topics:

- Migration Overview
- Why Migrate?
- Migration Scenarios

## Migration Overview

This document describes the processes required to fully migrate existing WebLogic Collaborate 1.0/1.0.1 and WebLogic Process Integrator 1.2/1.2.1 to WebLogic Integration 2.1. It also shows how to migrate WebLogic Integration 2.0 to WebLogic Integration 2.1.

WebLogic Integration provides the functionality businesses can use to develop new applications, integrate with existing systems, streamline business processes, and connect with trading partners. It enables intra- and inter-enterprise integration by providing functionality in the following areas:

- Application Integration
- Business Process Management (BPM)
- Business-to-Business Integration (B2B)

- Data Integration

Prior to this release the BPM was referred to as WebLogic Process Integrator (WLPI) and B2B was referred to as WebLogic Collaborate (WLC).

## Why Migrate?

BEA WebLogic Integration 2.1 uses a new database schema and security model. If you are using WebLogic Integration 2.0, WebLogic Process Integrator 1.2/1.2.1 or WebLogic Collaborate 1.0/1.0.1, you must migrate your existing database and security realm for compatibility with WebLogic Integration 2.1.

## Migration Scenarios

This document describes the migration procedures for the following scenarios:

- Migrating from WebLogic Process Integrator 1.2/1.2.1 to WebLogic Integration 2.1
- Migrating from WebLogic Collaborate 1.0/1.0.1 to WebLogic Integration 2.1.
- Migrating from WebLogic Integration 2.0 with no service pack or with service pack 1 to WebLogic Integration 2.1.
- Migrating from WebLogic Integration 2.0 service packs 2, 3, and 4 to WebLogic Integration 2.1.

# 2 Migrating from WebLogic Process Integrator 1.2/1.2.1

This section describes the processes required to fully migrate existing WebLogic Process Integrator 1.2/1.2.1 to WebLogic Integration 2.1. It contains information on the following subjects:

- Migration Process
- Before You Begin
- Migrating Your Database

## Migration Process

Migrating WebLogic Process Integrator 1.2/1.2.1 to WebLogic Integration 2.1 is a two-step process:

1. Convert the database schema to WebLogic Integration 2.1 format. To do this, you need to:
  - a. Convert the databases to 2.0 format, as described in “Migrating the Database Schema” on page 2-3.

- b. Convert the database schema from the 2.0 format to WebLogic Integration 2.1, as described in “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” in Chapter 5, “Migrating from WebLogic Integration 2.0 (SP 2, 3 and 4) to WebLogic Integration 2.1.”
2. Migrate the security realm, as described in “Migrating the Security Realm” on page 2-4. Migrating the security realm is required because WebLogic Integration 2.1 runs on a later version of WebLogic Server than did WebLogic Process Integrator 1.2/1.2.1.

**Note:** The WebLogic Integration 2.1 installation includes everything you need to migrate your databases and security realm. You do not need to have WebLogic Integration 2.0 installed to perform any of these migration steps.

After you have completed database migration and security realm migration, you will also need to assign permissions to previously existing users and roles. See “Assigning Permissions” on page 2-8 for more information.

Once the database is converted to the 2.1 level, tables that might not have existed in your WebLogic Process Integrator 1.2/1.2.1 database are added to WebLogic Integration. The result is that you now have the overall WebLogic Integration repository as opposed to just having the WebLogic Process Integrator tables.

## Before You Begin

Before attempting to migrate to WebLogic Integration 2.1, it is highly recommended that you back up the entire database and export all of your workflows. If migration fails due to an environmental error, you can fix the environment as necessary and remigrate. You can run the migration script as many times as necessary without causing any adverse affects.

## Migrating Your Database

This section includes information on the following subjects:

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

## Migrating the Database Schema

To migrate your database schema, use this procedure:

1. Using your database administration tool, back up your existing WebLogic Integration component database, as recommended in “Before You Begin” on page 2-2.
2. With the WebLogic component database open, execute the appropriate `.sql` file listed in Table 2-1. These files are located in `WLI_HOME/dbscripts/DB_TYPE/migrate` (where `DB_TYPE` is the subdirectory named for the type of database you are migrating; for example, Oracle or Sybase). Consult your database administrator for the procedure.

**Table 2-1 WebLogic Process Integrator Database Migration Scripts**

To migrate WebLogic component tables for this database ...	Use this <code>dbscript</code> 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>

3. Migrate the database schema from 2.0 format to WebLogic Integration 2.1 by running the `migratedb` script. See “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” in Chapter 5, “Migrating from WebLogic Integration 2.0 (SP 2, 3 and 4) to WebLogic Integration 2.1,” for instructions on running this script.

Once you have completed steps 1 through 3, 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 any existing users of your system, you can add them to the `WLPI_SUPER_USERS` variable in the `Migration.properties` file, or assign permissions in the Studio client application after migration. For information on 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 upon the version of WebLogic Process Integrator you are migrating and the security realm you have deployed, choose one of these migration procedures:

**Table 2-2 Migration Procedures**

<b>If you are migrating from . . .</b>	<b>Do this . . .</b>
WebLogic Process Integrator 1.2 or 1.2.1 and you deployed the security realm <code>com.bea.wlpi.rdbmsrealm.RDBMSRealm</code>	Use the procedure described in “Migrate from 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 security realm File Realm	Use the procedure described in “Migrate from File Realm (WebLogic Process Integrator 1.2.1 Only)” on page 2-7.

**Table 2-2 Migration Procedures**

<b>If you are migrating from . . .</b>	<b>Do this . . .</b>
WebLogic Process Integrator 1.2 and you deployed the <code>Weblogic.Properties</code> security realm	There is no automatic migration path. Please contact BEA Support for more information.

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

The following procedures assume knowledge of 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 the security realm `com.bea.wlpi.rdbmsrealm.RDBMSRealm`, use 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 and 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. In the right pane, select wlpiCachingRealm from the Caching Realm drop-down list.
11. In the left pane, Choose Services→JDBC→Connection Pools→wliPool and, in the right pane, select the Configuration and General tabs.
12. Verify that the database URL in 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 like this:

```
-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 setting *getGroupNewStatement* from *false* to *true* (that is, *getGroupNewStatement=true*). For the other database types, this value can be left *false*.

21. Shut down the WebLogic Server.
22. In the server startup script file, 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 `WLI_HOME/dbscripts/` directory and doing one of the following:
  - If you are using Windows NT, run the `bpm20_tutorial_enabler.cmd` script.
  - If you are using UNIX, run `bpm20_tutorial_enabler`.

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

The following procedures assume knowledge of 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, use this procedure:

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` and, 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 `fileRealm.properties` from the WebLogic Process Integrator 1.2.1 domain into the `fileRealm.properties` 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. 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`.

## Assigning Permissions

Once you have migrated your database and security realm, assign permissions to previously existing users and roles. Perform this task with the WebLogic Integration Studio. For information on default permission groups and their member users, and on assigning permissions to users and roles, see “[Administering Data](#)” in *Using the WebLogic Integration Studio*.

# 3 Migrating from WebLogic Collaborate 1.0/1.0.1

This section describes the processes required to migrate WebLogic Collaborate versions 1.0/1.0.1 to WebLogic Integration 2.1. It contains information on the following topics:

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

## Migration Process

The procedures in this section demonstrate that migrating WebLogic Collaborate 1.0/1.0.1 to WebLogic Integration 2.1 involves:

- Migrating the B2B repository from WebLogic Collaborate 1.0/1.0.1 to WebLogic Integration 2.1 format
- Migrating the necessary BPM-related entities like the security realm and workflows

- Migrating applications from the C-Enabler API of WebLogic Collaborate Releases 1.0/1.0.1 to the Messaging API of WebLogic Integration 2.1

Once WebLogic Collaborate has been migrated to WebLogic Integration 2.1, it will no longer exist as a single product; rather, it will become the Business-to-Business (or B2B) component of WebLogic Integration 2.1.

## Before You Begin

Before attempting to migrate to WebLogic Integration 2.1, it is highly recommended that you back up the entire database and export all of your workflows. If migration fails due to an environmental error, you can fix the environment as necessary and remigrate. You can run the migration script as many times as necessary without causing any adverse affects.

## Migrating the Repository and Workflows

WebLogic Integration 2.1 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 1.0/1.0.1 repository data file to a WebLogic Integration 2.1 repository data file.

## What Is Migrated?

In WebLogic Collaborate Releases 1.0/1.0.1, the configuration data is distributed among the following components:

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

The following procedure for migrating repository data achieves only a partial migration; it migrates data 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 Procedure

To migrate repository data and workflows, use this procedure:

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

Perform a full and 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 1.0/1.0.1. You can find these documents at:

[http://e-docs.bea.com/wlcollab/v1\\_0/chubag/bulkload.htm](http://e-docs.bea.com/wlcollab/v1_0/chubag/bulkload.htm)

or

[http://e-docs.bea.com/wlcollab/v1\\_0\\_1/chubag/bulkload.htm](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 WebLogic Sever 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 will be required for step 10.
3. Drop the WebLogic Collaborate 1.0-based tables.
4. Install WebLogic Integration 2.1 as described in [Installing BEA WebLogic Integration](#).
5. Run the `createdb` utility located in `WLI_HOME/bin`. Doing this creates the WebLogic Integration 2.1 database.

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 screen appears.

- b. Select Create Database and click Next.

The Database Selection screen appears.

- 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 will be 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 an end point value. The Bulk Migrator sets each end point value to `<placeholder>`. You must edit the new repository data file by setting each occurrence of `<placeholder>` to the correct end point value. The end point value specifies the trading partner end point. This end point is the URL for the trading partner.

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 ref value 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 1.0/1.0.1. You can find this document at one of the following URLs:

[http://edocs.bea.com/wlcollab/v1\\_0/cenbag/cfg.htm](http://edocs.bea.com/wlcollab/v1_0/cenbag/cfg.htm)

or

[http://edocs.bea.com/wlcollab/v1\\_0\\_1/cenbag/cfg.htm](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 Releases 1.0/1.01 and the repository data elements for WebLogic Integration 2.1.

**Table 3-1 Relationships Between Repository Data Elements**

<b>WebLogic Collaborate Releases 1.0/1.01</b>	<b>WebLogic Integration 2.1</b>
Business protocol	Transport and end point
Business protocol definition	Business protocol definition
C-hub	WebLogic Collaborate
Collaborator	Collaboration agreement
Conversation definition	Conversation definition
C-space	Delivery channel
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.

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 1.0/1.0.1. You can find these documents at:

[http://e-docs.bea.com/wlcollab/v1\\_0/chubag/bulkload.htm](http://e-docs.bea.com/wlcollab/v1_0/chubag/bulkload.htm)

or

### 3 Migrating from WebLogic Collaborate 1.0/1.0.1

---

[http://e-docs.bea.com/wlcollab/v1\\_0\\_1/chubag/bulkload.htm](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 1.0/1.01 c-hub, the old c-hub is displayed in the B2B Console for WebLogic Integration 2.1 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 in: `WLI_HOME/bin` (UNIX) or `WLI_HOME\bin` (Windows NT). At the command prompt, enter the command appropriate for your platform:

- If you are using Windows NT, enter:

```
PIMigrator.cmd workflow-1.0.xml workflow-2.0.xml
```

- If you are using UNIX, enter:

```
../PIMigrator.sh workflow-1.0.xml workflow-2.0.xml
```

In the preceding command lines:

- `workflow-1.0.xml` represents the full pathname of the workflow created with WebLogic Collaborate Version 1.0. `workflow-1.0.xml` was generated in step 2.
  - `workflow-2.0.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 WebLogic Studio. From there, 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 2.1 to correctly reflect the new concepts and terminology of the B2B component of WebLogic Integration architecture. The 1.0/1.0.1 C-Enabler API is deprecated but still supported. It is recommended that you use the new Messaging API for Java applications. Please refer to [Programming Messaging Applications for B2B Integration](#) and the WebLogic Integration 2.1 Javadoc for details about the new messaging API.

The following sections describe how to migrate applications from the C-Enabler API (for WebLogic Collaborate Releases 1.0 /1.0.1) to the still-supported C-Enabler API (for WebLogic Integration 2.1). This section includes information on the following subjects:

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

## Introduction

**Note:** If you are migrating a Java messaging application that was written using the WebLogic Collaborate C-Enabler API to WebLogic Integration 2.1, the migrated application must be run in a separate JVM in nonpersistent mode.

XOCP is the default protocol used by WebLogic Integration for exchanging business messages. In WebLogic Collaborate Releases 1.0 /1.0.1, XOCP messaging applications were written using the public XOCP C-Enabler application program interface (API). A message was packaged in an XOCP message envelope and associated with various message headers that specified how to deliver and process the

message. In WebLogic Integration 2.1, the c-enabler and c-hub entities have been integrated. The integrated product is known simply as B2B integration component of WebLogic Integration.

The WebLogic Integration Messaging API now supports the C-Enabler functionality, but 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 provided for developers who want to maintain their installations using a Java program as the interface. 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 the collaborative conversations that are taking place.

When a business process initiates a specific task (for example, issuing a request for a quote), a central management point is created for the 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 WebLogic Collaborate Releases 1.0 /1.0.1 XOCP applications for migration purposes. The configuration characteristics of your WebLogic Integration installation will determine whether functionality previously associated with the c-enabler or the c-hub is invoked. Table 3-2 summarizes how the functionality is mapped.

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

<b>Collaborate 1.0/1.0.1 Term</b>	<b>WebLogic Integration 2.1 Term</b>	<b>Definition</b>
Attachment	Attachment	Non XML payload part of a business message; for example, a GIF file.
Business Document	Business Document	XML-based payload part of the business message.
Business Message	Business Message	A multi part MIME message, consisting of one or more payload parts. A part can be an XML document (a business document) or non XML file (an attachment).
Business Protocol Definition	Business Protocol Definition	The definition of a business transaction protocol, for example, Name=XOCP.

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

<b>Collaborate 1.0/1.0.1 Term</b>	<b>WebLogic Integration 2.1 Term</b>	<b>Definition</b>
Collaborator	Trading Partner Session	In WebLogic Collaborate Releases 1.0/1.0.1, 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 communicate 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, for example, a purchase order.
Conversation Type	Not applicable	There is no corresponding term for conversation type in WebLogic Integration 2.1. In WebLogic Collaborate Releases 1.0 /1.0.1, conversation type defined a conversation for a trading partner. It consisted of the following properties: <ul style="list-style-type: none"> <li>■ Conversation name (for example, purchase order)</li> <li>■ Version</li> <li>■ Role of a trading partner in that conversation (for example, buyer or seller)</li> </ul>
C-Space	Delivery Channel	A grouping of Trading Partner, Subscription, and Business Protocol.
C-enabler Node	Spoke	A trading partner communicates with others with the help of a hub (intermediary).

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

<b>Collaborate 1.0/1.0.1 Term</b>	<b>WebLogic Integration 2.1 Term</b>	<b>Definition</b>
C-enabler API	WebLogic Integration Messaging API	<p>XOCP applications that used the C-enabler API can be migrated to use the WebLogic Integration Messaging API.</p> <p>The C-enabler API was the interface between the XOCP application and WebLogic Collaborate at run time. This API was used, for example, to start c-enabler sessions, create and participate in conversations, and send and receive business messages.</p>
C-enabler Session	Trading Partner Session	<p>In WebLogic Collaborate Releases 1.0 /1.0.1, a c-enabler session is a logical session between a c-enabler node and one c-hub for one particular c-space. In WebLogic Integration 2.1, a trading partner session occurs in the delivery channel.</p>
C-hub	Intermediary	<p>A hub is a WebLogic Server running a set of shared services. For example:</p> <ul style="list-style-type: none"> <li>■ Repository</li> <li>■ Administration</li> <li>■ Conversation coordination, message routing, and mediation</li> </ul>
Message	Message	<p>A data entity that is used to communicate between hubs and spokes. It is a multi part MIME message.</p>

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

<b>Collaborate 1.0/1.0.1 Term</b>	<b>WebLogic Integration 2.1 Term</b>	<b>Definition</b>
Role	Role	Each conversation has two or more roles, for example, buyer and seller. A role is defined in terms of the documents that can be sent or received by a collaborator who has assumed that role in the conversation.
Send(Publish)-Side filter	Send(Publish)-Side filter	When sending a document, this filter (X-Path) is specified by an application to determine the list of recipient trading partners. For example: <pre>//trading partner[@name="Bea"]</pre>
Trading Partner	Trading Partner	A representation of an entity, such as a company, that wants to participate in one or more delivery channels.

## Supporting the WebLogic Collaborate 1.0/1.0.1 C-Enabler API

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

To support migration to WebLogic Integration 2.1, a WebLogic Collaborate 1.0/1.0.1 C-Enabler application must have existing database support, and all of its trading partner collaboration agreements must be predefined in the repository. In order to collaborate in business transactions using the XOCP protocol, the application must be aligned with the WebLogic Integration Messaging API to achieve as much backward compatibility as possible. Changes are required in the following areas in order for the existing XOCP applications to migrate to the WebLogic Integration Messaging API:

- C-Enabler Configuration
- Repository

### C-Enabler Configuration

In WebLogic Collaborate Releases 1.0 /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 the conversations with other trading partners.

In WebLogic Integration 2.1, the identification and configuration of a trading partner is predefined 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 will work 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 predefined in the repository. To migrate WebLogic Collaborate Release 1.0/1.0.1 data repository to WebLogic Integration 2.1, use the Bulk Migrator utility described in “Migrating the Repository and Workflows” on page 3-2.

# WebLogic Collaborate 1.0/1.0.1 C-Enabler API Implementation

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

## 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`

## Interfaces

- `com.bea.b2b.EnablerSession`
- `com.bea.b2b.enabler.xocp.XOCPEnablerSession`
- `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`

## 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.MinParser`
- `com.bea.b2b.protocol.xocp.messaging.Ping`
- `com.bea.b2b.protocol.xocp.messaging.QualityOfService`
- `com.bea.b2b.protocol.xocp.messaging.XOCPMessage`
- `com.bea.b2b.protocol.xocp.messaging.XOCPMessageToken`



# 4 Migrating from WebLogic Integration 2.0 (No SP or SP 1) to WebLogic Integration 2.1

This section describes the procedure for migrating from BEA WebLogic Integration 2.0 with no service pack or with service pack 1 to BEA WebLogic Integration 2.1.

Migrate WebLogic Integration 2.0 with no service pack or with service pack 1 to WebLogic Integration 2.1 by completing the following steps:

1. Migrate from WebLogic Integration 2.0 with no service pack or with service pack 1 to WebLogic Integration 2.0 service pack 4 by following the directions in “Migration Issues” in *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](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 of being a BEA Customer Support contract customer. More information on BEA WebSUPPORT is available at the following URL:

## 4 Migrating from WebLogic Integration 2.0 (No SP or SP 1) to WebLogic Integration 2.1

---

<http://www.bea.com/support/web.html>

2. Migrate from WebLogic Integration 2.0 service pack 4 to WebLogic Integration 2.1 by following directions in Chapter 5, “Migrating from WebLogic Integration 2.0 (SP 2, 3 and 4) to WebLogic Integration 2.1.”

# 5 Migrating from WebLogic Integration 2.0 (SP 2, 3 and 4) to WebLogic Integration 2.1

This section describes the procedure for migrating from BEA WebLogic Integration 2.0 service packs 2, 3, and 4 to BEA WebLogic Integration 2.1. It contains information on the following subjects:

- Overview
- Before You Begin
- Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode
- Running the Migration Script Manually

# Overview

If you are currently running WebLogic Integration 2.0 Service Pack 2, 3 or 4, you can migrate to WebLogic Integration 2.1 by following the migration steps described in “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” on page 5-4 or “Running the Migration Script Manually” on page 5-10.

The migration steps described “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” on page 5-4 or “Running the Migration Script Manually” on page 5-10 can also be used to complete the overall migration processes described in Chapter 2, “Migrating from WebLogic Process Integrator 1.2/1.2.1,” and Chapter 3, “Migrating from WebLogic Collaborate 1.0/1.0.1,” or Chapter 4, “Migrating from WebLogic Integration 2.0 (No SP or SP 1) to WebLogic Integration 2.1.” 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.

## Before You Begin

Before following the migration steps described in “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” on page 5-4 or “Running the Migration Script Manually” on page 5-10, you must consider the following before proceeding:

- Domain Migration
- Remote Database Server
- Security File Realms

---

## Domain Migration

The migration steps described in “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” on page 5-4 and “Running the Migration Script Manually” on page 5-10 will only migrate the WebLogic Integration repository information stored in database tables. These steps will not migrate your application WebLogic Server domain information.

To migrate your application WebLogic Server domain information complete the following steps:

1. Migrate the WebLogic Integration 2.0 repository information into one of the preconfigured WebLogic Integration 2.1 domains as described in “Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode” on page 5-4 and “Running the Migration Script Manually” on page 5-10. For more information on domains see “WebLogic Server Domains” in “[Getting Started](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.
2. Copy any application specific entries from your existing WebLogic Integration 2.0 domain’s `config.xml` file to the WebLogic Integration 2.1 domain’s `config.xml` file.

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

## Remote Database Server

If the machine where WebLogic Integration 2.1 is installed and the migration utility will be run is different from the machine which hosts the database server with the WebLogic Integration repository, the following prerequisite must be satisfied before running the migration utility:

- A client installation of the database product must be installed on the WebLogic Integration machine.
- The database directory which contains the specific command line executable specific for the database must be included in the `PATH` environment variable. For

example for the Oracle database on UNIX, `$ORACLE_HOME/bin` must be included in the `PATH` environment variable.

### Security File Realms

Before beginning either the graphical or manual migration processes, 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`. This procedure is required only if you are using the file realm for security.

# Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode

To migrate to WebLogic Integration 2.1 using the WebLogic Integration database configuration wizard in graphical mode, complete the following steps:

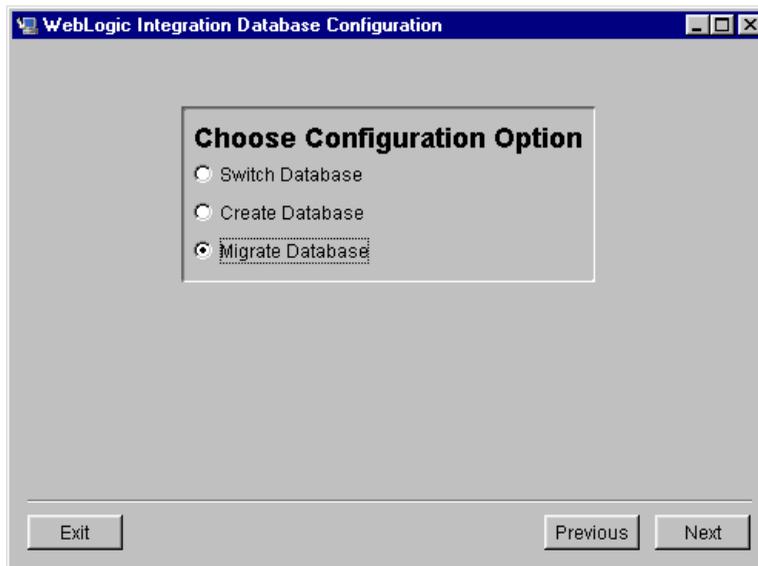
1. Start the WebLogic Integration database configuration wizard by completing the procedure appropriate for your platform:
  - Windows:  
Choose Start→BEA WebLogic E-Business Platform→WebLogic Integration 2.1→Configure.
  - UNIX:
    - a.) Execute the following commands:

```
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.

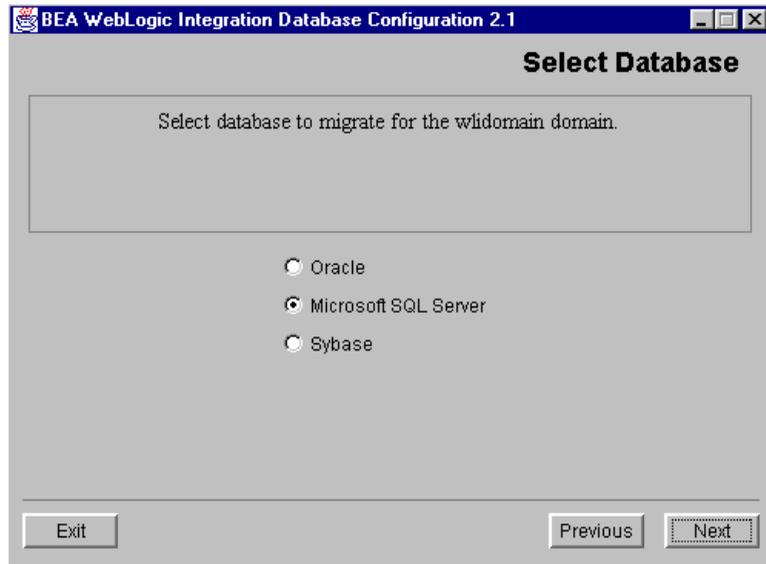
c.) Select a domain, and then click Next. For more information on domains see “WebLogic Server Domains” in “[Getting Started](#)” in *Starting, Stopping, and Customizing BEA WebLogic Integration*.

The Choose Configuration Option dialog box is displayed.



2. Select Migrate Database and select Next.

The Select Database dialog box appears.



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

**Note:** Migration is not available for Cloudscape.

The Configure *DB\_TYPE* Database dialog box appears (*DB\_TYPE* is the database you selected, in this example Microsoft SQL).

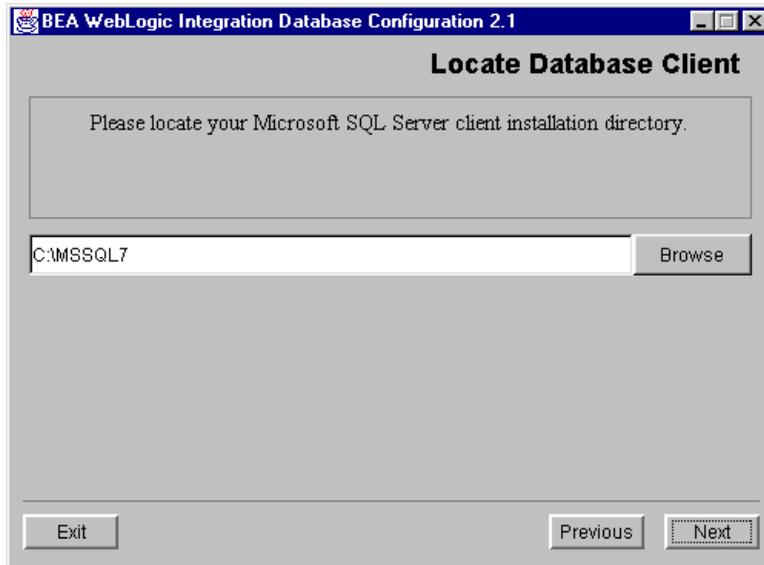
The screenshot shows a graphical user interface window titled "BEA WebLogic Integration Database Configuration 2.1". The main heading is "Configure Microsoft SQL Database". The window contains a form with the following fields and values:

Database Host Name:	EDAL
Database Port:	1433
Database User:	sa
Database Password:	
Database Name:	WLIDB
Server Name:	EDAL

Below the form is a "Reset" button. At the bottom of the window, there are three buttons: "Exit", "Previous", and "Next". The "Next" button is highlighted with a dashed border, indicating it is the active option.

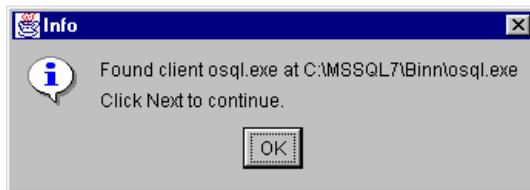
4. Depending upon the choice of the database, you will be prompted for database related information; for example, host, port, user, password, service name. Fill out this dialog box and select Next.

The Locate Database Client dialog box appears.



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

The database found confirmation box appears.

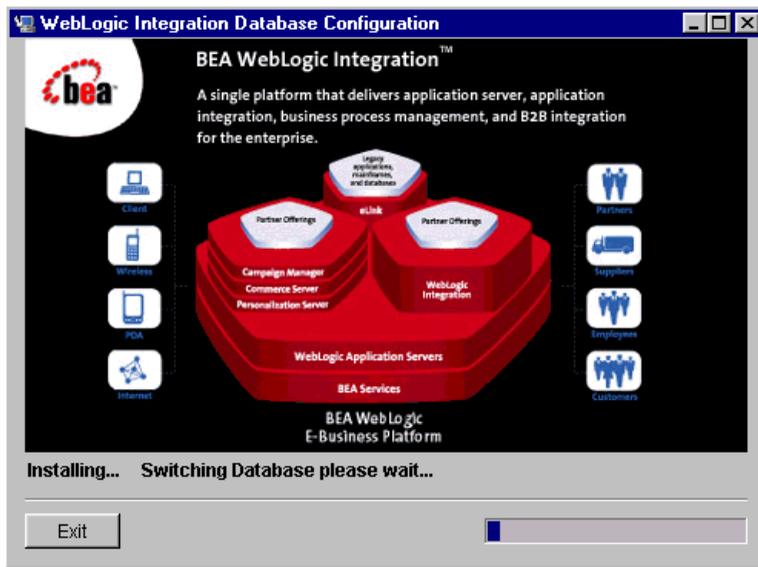


**Note:** This confirmation box appears immediately after step 6, and overlays the Locate Database Client screen, if you do not need to locate your client installation directory.

6. Select **OK** and then select **Next** on the Locate Database Client screen.

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

## Migrating Using the WebLogic Integration Database Configuration Wizard in Graphical Mode



This dialog box stays visible until the progress meter in the lower right corner is completely filled. Finally, the Changes Successful dialog box appears.



7. Select Finish to end the migration process.

This completes the WebLogic Integration 2.0 to WebLogic Integration 2.1 migration process.

# Running the Migration Script Manually

## Windows

Run the migration script manually on Windows, by completing the following steps:

1. Go to the bin directory in your WebLogic Integration installation 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 with the domain selected in the previous step with the appropriate path based on your installation. For example:

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

4. Run the `switchdb` script with the appropriate database for the domain specified in the previous step. Valid options are `oracle` or `sqlserver`. The database specified here should match the database selected for the domain during installation. The database selected during installation for your production environment sets the default database for the `bpmdomain`, `eaidomain`, and `wlidomain` domains. The database selected during installation for your sample environment sets the default database for the `samples` domain. For example, if during the installation you select the Oracle database for production domain, and now you are migrating to the WebLogic Integration 2.1 `wlidomain` domain, then specify the `oracle` option for `switchdb` command:

```
switchdb oracle
```

5. Run the migration script to migrate the WebLogic Integration 2.0 (SP2, SP3, or SP4) repository data to the WebLogic Integration 2.1 domain:

```
migratedb
```

## UNIX

Run the migration script manually on UNIX, by completing the following steps:

1. Go to the WebLogic Integration home's bin 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 with the domain selected in the previous step with the appropriate path based on your installation. For example:

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

4. Run the `switchdb` script with the appropriate database for the domain specified in the previous step. The database specified here should match the database selected for the domain during installation. The database selected during installation for your production environment sets the default database for the `bpmdomain`, `eidomain`, and `wlidomain` domains. The database selected during installation for your sample environment sets the default database for the `samples` domain. For example, if during the installation you select the Oracle database for production domain, and now you are migrating to the WebLogic Integration 2.1 `wlidomain` domain, then specify the `oracle` option for `switchdb` command:

```
switchdb oracle
```

5. Edit the `migrateCommonRep` script and remove the `-v` option from the following line:

```
$JAVA_HOME/bin/java $DB_JVMARGS $JAVA_OPTIONS -classpath $WLICP  
com.bea.eci.migration.Migrate -v $CTLFIL
```

6. Edit the `migratedb` script and remove the `.sh` from the following line:

```
.$WLI_HOME/bin/migrate.sh || exit 1
```

7. Run the migration script to migrate the WebLogic Integration 2.0 (SP2, SP3, or SP4) repository data to the WebLogic Integration 2.1 domain:

```
migratedb
```