



# BEA AquaLogic Data Services Platform™

## Installation Guide

Note: Product documentation may be revised post-release and made available from the following BEA e-docs site:

<http://e-docs.bea.com/al dsp/docs30/index.html>

Versions: 3.0, 3.01  
Revised: April 2008



# Contents

<b>1. Preparing to Install ALDSP</b>	
Installation Overview .....	1-1
Supported Configurations .....	1-3
Installation Prerequisites .....	1-3
ALDSP Installation Components .....	1-3
Preconfigured Samples Domain .....	1-4
Internationalization Support .....	1-5
<b>2. Installing ALDSP Using GUI Mode</b>	
Before You Install .....	2-1
Installing Using GUI Mode .....	2-2
<b>3. Installing ALDSP Using Console or Silent Mode</b>	
Before You Install .....	3-2
Installing Using Console Mode .....	3-2
Using Silent Console Mode to Install ALDSP .....	3-5
Using Silent Mode on Windows and UNIX Systems .....	3-6
Exploring the Silent Mode Installer Properties File .....	3-8
<b>4. Post-Installation Tasks</b>	
Verifying the Installation .....	4-2
Step 1: Start Data Services Studio .....	4-2
Step 2: Configure the ALDSP-enabled Server Environment .....	4-2

Step 3: Start the Server . . . . .	4-4
Starting the Development of a Data Integration Solution . . . . .	4-4

## 5. Uninstalling ALDSP

Automated Uninstallation . . . . .	5-1
Manual Uninstallation . . . . .	5-2

## 6. Upgrading from ALDSP 2.5 to ALDSP 3.0

Upgrading from ALDSP 3.0 to 3.01 . . . . .	6-1
Upgrading from ALDSP 2.5 Prerequisites . . . . .	6-2
Upgrading ALDSP 2.5 Artifacts to ALDSP 3.0 . . . . .	6-2
Upgrading an ALDSP 2.5 Application . . . . .	6-3
Upgrading an ALDSP Control from ALDSP 2.5 to 3.0 . . . . .	6-8
Importing ALDSP 2.5 Configuration to 3.0 . . . . .	6-13
Post-Upgrade Artifact Mapping in ALDSP 3.0 . . . . .	6-15
Post-Upgrade Tasks for Java Web Service . . . . .	6-20
ALDSP 2.5 Upgrade: Known Issues and Workarounds . . . . .	6-21

# Preparing to Install ALDSP

Before you install ALDSP, you need to verify that your system meets minimum requirements. This chapter provides information about supported platforms, system hardware and software requirements, and describes installation prerequisites. It also provides an overview of the ALDSP components that get installed.

The following sections are included:

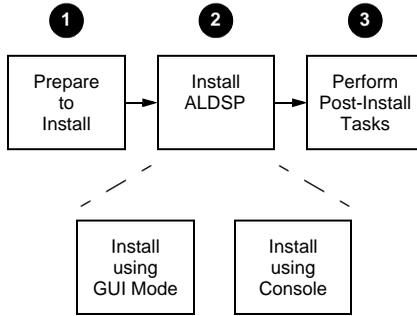
- [Installation Overview](#)
- [Supported Configurations](#)
- [Installation Prerequisites](#)
- [ALDSP Installation Components](#)

**Note:** ALDSP was initially named Liquid Data. Some instances of the original name remain in the product, installation path, and components.

## Installation Overview

This section provides a high-level overview of the steps required to install ALDSP, illustrated in [Figure 1-1](#).

**Figure 1-1 Overview of Installation Process**



### **Step 1: Prepare to install**

1. Preparing to install ALDSP involves checking the supported platforms and databases, and verifying that you have the installation prerequisites in place. You can also familiarize yourself with a description of what gets installed on your system. To prepare to install the software, complete the remaining sections of this chapter.
2. Uninstall any previous versions of ALDSP. For details see [Chapter 5, “Uninstalling ALDSP.”](#)

**Note:** The Uninstall option does not remove any ALDSP or other Workshop projects that you have created.

### **Step 2: Install ALDSP 3.0**

You can install ALDSP either in GUI mode, featuring an easy to use graphical installer application, or in interactive Console mode, suitable on UNIX systems without a graphics (windowing) workstation. Alternatively, you can choose to install ALDSP on Windows and UNIX systems using Silent mode, which reads an installer properties file to determine the install options.

For more information about using GUI mode, see [Chapter 2, “Installing ALDSP Using GUI Mode.”](#) For more information about using Console mode and Silent mode, see [Chapter 3, “Installing ALDSP Using Console or Silent Mode.”](#)

### **Step 3: Perform post-installation tasks**

After installation, you can verify that the installation was successful, and explore Windows shortcuts and UNIX paths to key components. For more information, see [Chapter 4, “Post-Installation Tasks.”](#)

## Supported Configurations

Supported operating systems, DBMS vendors, and other configurations are described under [Supported Configurations](#).

## Installation Prerequisites

Additional software needs to be installed with ALDSP including:

A supported version of WebLogic Server 9.2, Workshop for WebLogic 9.2 (WebLogic Platform 9.2) or Eclipse 3.2.2/WTP 1.5.4.

Latest JRockit version ([JRockit version 5.0 R27.4 JDK](#)) to ensure higher ALDSP performance.

**Note:** ALDSP 3.0 includes Data Services Studio, which can run either under Eclipse 3.2.2/WTP 1.5.4 or Workshop for WebLogic 9.2 or 10.1.

By default, Workshop for WebLogic 9.2 and Eclipse version 3.2.2 is installed with ALDSP. However, if you have previously installed versions of Workshop for WebLogic 9.2 or Eclipse 3.2.2 and want to continue using that instance to work with ALDSP, then specify the Workshop for WebLogic 9.2 or Eclipse home directory path during installation. ALDSP installer will use the same instance for installation.

WebLogic Platform 9.2 components may be obtained from the BEA Download Center:

<http://commerce.bea.com/downloads/products.jsp>

**Note:** If you are using Windows 2000, the maximum classpath size can be exceeded. For this reason it is recommended that when you install ALDSP, you install it into `<beahome>` directory with a directory name of four characters or less such as `<bea>`.

## ALDSP Installation Components

[Table 1-1](#) provides a list of components that get installed after you successfully complete installing ALDSP.

**Table 1-1 ALDSP Installation Components**

Component Name	Sub-components
ALDSP Runtime  <b>Note:</b> Installing the runtime is mandatory for ALDSP and is a prerequisite for installing other components.	<ul style="list-style-type: none"> <li>• ALDSP runtime including the XQuery engine</li> <li>• Configuration framework</li> <li>• ALDSP Administration Console</li> <li>• ALDSP base domain template</li> </ul> <p><b>Note:</b> WebLogic Server 9.2 MP2 is a prerequisite for ALDSP runtime.</p>
ALDSP IDE and Client Tools	<ul style="list-style-type: none"> <li>• Common workshop 10.2 plugins</li> <li>• ALDSP IDE plugins</li> <li>• ALDSP Microsoft Excel add-in</li> <li>• Workshop for Weblogic 9.2 common components</li> <li>• AquaLogic tooling common components including AquaLogic Sync and AquaLogic Enterprise Registry browser.</li> </ul> <p><b>Note:</b> Data Services Studio (Eclipse 3.2.2/WTP 1.5.4) or Workshop for WebLogic 10.2 is a prerequisite for ALDSP tools.</p>
ALDSP Samples	<ul style="list-style-type: none"> <li>• Samples domain</li> <li>• Sample project predeployed in the samples domain.</li> <li>• Retail Dataspace Sample Application</li> </ul>

When you install ALDSP, a valid evaluation license is automatically included with the installation.

## Preconfigured Samples Domain

The full ALDSP installation provides a preconfigured samples domain, as shown in [Table 1-2](#).

**Table 1-2 ALDSP Samples Preconfigured Domain and Start Commands for Samples Server**

Platform	Windows and UNIX Paths to Start in Each Domain	Description
Windows	Start > All Programs > <BEA Products> > Examples > Start Examples Server  or  Run startWebLogic.cmd from any one of the following locations: <bea_home>\aldsp_3.0\samples\domains\aldsp\  or <bea_home>\aldsp_3.0\samples\domains\aldsp\bin	Starts the examples server on Windows  Starts the ALDSP samples server on Windows
UNIX	Run the startWebLogic.sh command from any of the following locations: <bea_home>/aldsp_3.0/samples/domains/aldsp/  or <bea_home>/aldsp_3.0/samples/domains/aldsp/bin	Starts the ALDSP examples server on UNIX

**Note:** <bea\_home> is the home directory for installing ALDSP and <BEA Products> is the root location of the ALDSP installation.

For a detailed explanation of domains, see [“Creating WebLogic Configurations Using the Domain Control Wizard”](#) in the WebLogic Platform documentation.

## Internationalization Support

ALDSP is internationalized and supports multi-byte data from the underlying data sources. Specifically, ALDSP works with Japanese character sets, where the underlying databases are running in Japanese locales.

## Preparing to Install ALDSP

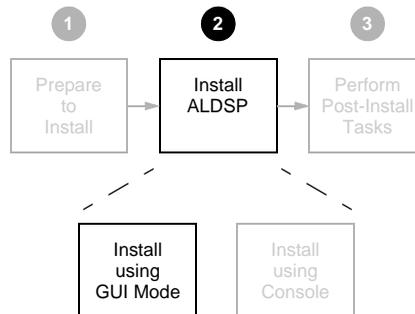
# Installing ALDSP Using GUI Mode

This chapter describes how to install ALDSP using GUI mode. It includes the following sections:

- [Before You Install](#)
- [Installing Using GUI Mode](#)

Figure 2-1 illustrates the current step in the overall installation process.

**Figure 2-1 GUI Mode Installation Step**



## Before You Install

Before you begin installing ALDSP, confirm that the following prerequisites are met:

- A supported version of WebLogic Server 9.2 is installed in the <bea\_home> directory in which you are going to install ALDSP.

- Any previous installations of ALDSP have been uninstalled. For details see [Chapter 5, “Uninstalling ALDSP.”](#)
- It is recommended for Windows (and a requirement for UNIX and Linux) that an entry for the JDK 150\_10 bin directory (for example: <bea\_home>/JDK 150\_10/bin) be included in your PATH environment variable setting before any other JDK bin directories.
- By default, Data Services Studio (Eclipse 3.2.2) is available for installation with ALDSP 3.0 in the <bea\_home>\aldsp\_3.0\tools directory. If you are planning to use an Eclipse version that is not bundled with ALDSP, then you need to manually clean up the Eclipse configuration directory to avoid some known issues in Eclipse 3.2.2. All files except `config.ini` and files under `.settings` subdirectory should be deleted.

For additional information about installation prerequisites, see “[Installation Prerequisites](#)” on page 1-3 in [Chapter 1, “Preparing to Install ALDSP.”](#)

## Installing Using GUI Mode

This section describes how to install ALDSP using the GUI mode on Microsoft Windows and supported UNIX and Linux-based platforms.

To install ALDSP in GUI mode:

1. Launch the GUI installer.

On Microsoft Windows-based systems:

- a. Navigate to the folder to which you downloaded the ALDSP installer using Windows Explorer.
- b. Double-click the installer executable file. For example:

```
aldsp_300_win32.exe
```

On UNIX and Linux-based systems, do the following:

- a. Verify that the console attached to the machine on which you are installing the software supports a Java-based GUI.
- b. Open a command window, and change (`cd`) to the directory to which you downloaded the ALDSP installer file.
- c. Start the installer in a new shell by entering the following:

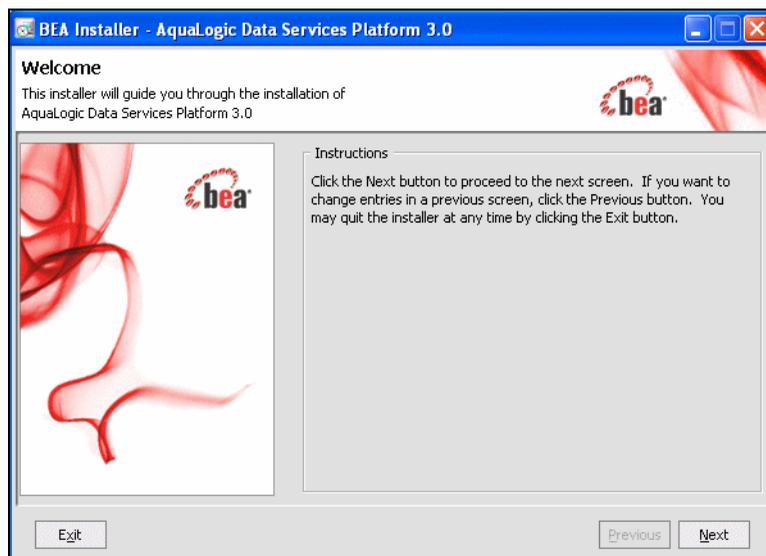
```
sh filename.bin
```

where *filename.bin* is the name of the ALDSP installation program specific to your platform. For example, enter the following to start the Solaris version of the ALDSP installation program:

```
sh aldsp_300_solaris32.bin
```

The installer application welcomes you to install ALDSP, as illustrated in [Figure 2-2](#).

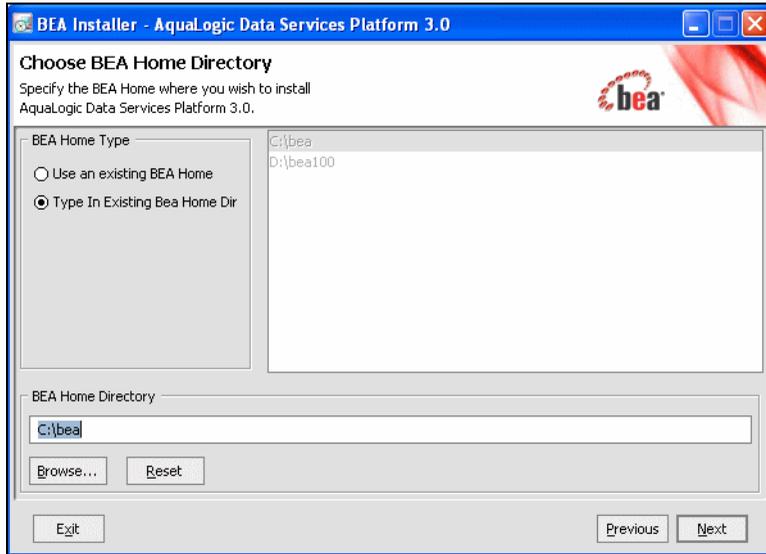
**Figure 2-2 ALDSP Installation Welcome Screen**



2. Click Next to begin the installation. The License Agreement appears.
3. Review the License Agreement, using the scroll bar to display the text. Choose Yes to accept, and click Next.

A screen appears enabling you to specify the BEA home directory, as illustrated in [Figure 2-3](#).

**Figure 2-3 Choose BEA Home Directory**



The BEA home directory (<bea\_home>) serves as the central support directory for all the BEA products installed on your system.

4. Specify the BEA home directory, and click Next.

You can choose a directory from the list, or type the directory name in the BEA Home Directory field. You must install ALDSP in the same home directory where you installed WebLogic Platform 9.2

A warning message is displayed under two conditions:

- When a previously installed copy of ALDSP is found. In this case, you must uninstall ALDSP before installing the product again. (For details see [Chapter 5, “Uninstalling ALDSP.”](#))
- When WebLogic Platform 9.2 is not found in the specified BEA home directory. Verify that the specified directory has a WebLogic Platform 9.2 installation. If not, exit the ALDSP installation and install WebLogic Platform 9.2 before proceeding.

A screen appears enabling you to choose the type of installation, as illustrated in [Figure 2-4](#).

**Figure 2-4 Choose Install Set**

5. Choose either Complete or Custom as the install type, and click Next.

The following describes the options:

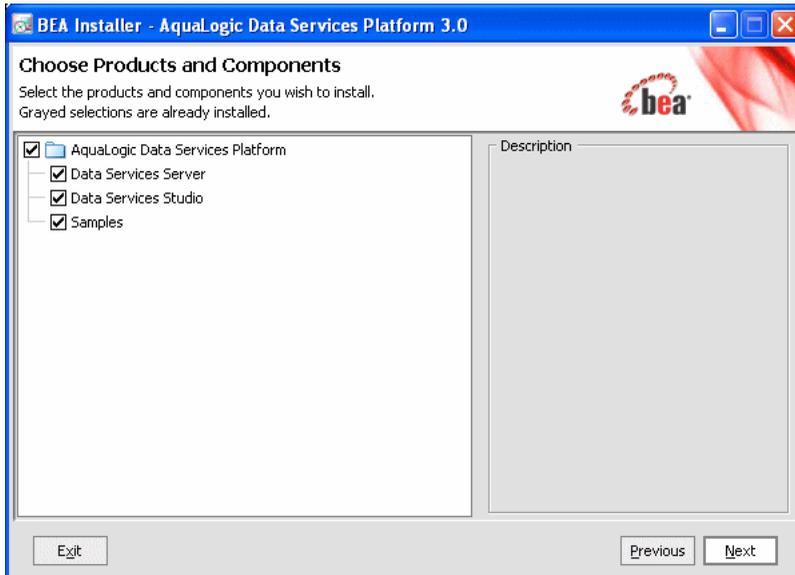
- Complete

This option installs all ALDSP components mentioned in [ALDSP Installation Components](#).

- Custom

Enables you to choose the ALDSP components you need to install. If you select this option and click Next, the screen shown in [Figure 2-5](#) is displayed.

**Figure 2-5 Choosing Components to Install**



6. Choose the components and click Next. This screen allows you to specify the directory where you want to install ALDSP. By default, ALDSP is installed in `<bea_home>\aldsp_3.0`.

The installer displays a progress screen, and begins installing the files on your system. When the installation is complete, the Installation Complete screen appears.

7. Click Done.

You have successfully installed an instance of ALDSP.

# Installing ALDSP Using Console or Silent Mode

This section describes how to install AquaLogic Data Services Platform using either Console mode or Silent mode. Console mode is an interactive installation that you can use on UNIX systems without a graphics (windowing) workstation.

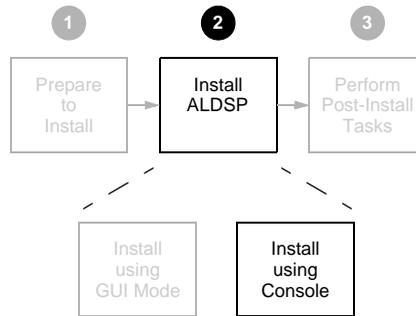
Silent mode is a non-interactive installation on a Windows or UNIX system. In Silent mode, the installer application uses a properties file to obtain installation parameters.

The chapter includes the following sections:

- [Before You Install](#)
- [Installing Using Console Mode](#)
- [Using Silent Console Mode to Install ALDSP](#)

Figure 3-1 illustrates the current step in the overall installation process.

Figure 3-1 Console Mode Installation Step



## Before You Install

Before you begin installing ALDSP, confirm that the following conditions are met:

- A supported version of WebLogic Server or WebLogic Platform 9.2 is installed in the <bea\_home> directory in which you are installing ALDSP.
- ALDSP is not running on the system.

You must remove ALDSP in cases when it is located in the same <bea\_home> directory to which you plan to install. For details see [Chapter 5, “Uninstalling ALDSP.”](#)

- An entry for the JDK 150\_10 bin directory (for example <bea\_home>/JDK\_150\_10/bin) must be included in your PATH environment variable setting before any other JDK bin directories.
- By default, Data Services Studio (Eclipse 3.2.2) is available for installation with ALDSP 3.0 in the <bea\_home>\aldsp\_3.0\tools directory. If you are planning to use an Eclipse version that is not bundled with ALDSP, then you need to manually clean up the Eclipse configuration directory to avoid some known issues in Eclipse 3.2.2. All files except config.ini and files under .settings subdirectory should be deleted.

For additional information about installation prerequisites, see [“Installation Prerequisites” on page 1-3 in Chapter 1, “Preparing to Install ALDSP.”](#)

## Installing Using Console Mode

This section describes the console-mode installation procedure.

**Note:** You cannot install ALDSP in console mode on Solaris systems running in a Japanese locale. In cases when you are installing on a Japanese Solaris system, change your locale in the shell in which you are installing to English, and then install the product. After ALDSP is installed, you can run it in a Japanese locale.

To install ALDSP on a UNIX system in console mode:

1. Change to the directory (cd) that contains the ALDSP installer.
2. Start the installation program.

The installer file name has the following form:

```
filename.bin
```

where *filename* is the name of the ALDSP installation program specific to your platform. For example, enter the following to start the Solaris version of the ALDSP installation program:

```
sh aldsp_300_solaris32.bin -mode=console
```

Similarly, enter the following to start the Linux version of the ALDSP installation program:

```
./aldsp_300_linux32.bin -mode=console
```

A message appears indicating that the software is being extracted, followed by a welcome message, as shown in the following text:

```
<----- BEA Installer - BEA AquaLogic Data Services Platform 3.0
----->
Welcome:
-----

This installer will guide you through the installation of BEA AquaLogic
Data Services Platform 3.0. Type "Next" or enter to proceed to the next
prompt. If you want to change data entered previously, type "Previous".
You may quit the installer at any time by typing "Exit".

Enter [Exit][Next]>
```

3. Press Enter to continue.

The license agreement is displayed. Review the entire agreement. When you reach the end of the agreement, you are prompted to accept or reject the terms.

```
Select Option:
1 - Yes, I agree with the terms of the license
2 - No, I do not agree with the terms of the license
```

## Installing ALDSP Using Console or Silent Mode

```
Enter option number to select OR [Down][Exit][Previous]>
```

4. Enter 1 if you accept the license agreement.

You are prompted to choose the BEA home directory, as shown in the following text:

```
<----- BEA Installer - AquaLogic Data Services Platform 3.0
----->
Choose BEA Home Directory:
-----

"BEA Home" = [/usr/local/bea]

Input existing BEA Home OR [Exit][Previous][Next]>
```

The <bea\_home> directory serves as the central support directory for all the BEA products installed on your system. For a detailed description of how this directory is used, see “BEA Home Directory” in “Preparing to Install WebLogic Server” in [Installing BEA WebLogic Server](#).

A warning message is displayed in cases when a previous installation of ALDSP is found, or when WebLogic Server 8.1 is not found in the specified BEA home directory.

5. Press Enter to accept the default BEA home directory, or enter an alternative directory.

When prompted, confirm the directory you selected. You are prompted to choose whether you want to perform a complete or custom installation, as shown in the following text:

```
<----- BEA Installer - AquaLogic Data Services Platform 3.0
----->
Choose Install Type:
-----

->1|Complete
|Install the complete BEA AquaLogic Data Services Platform 3.0.
2|Custom Installation
|Choose software components to install and perform optional
configuration.

Enter index number to select OR [Exit][Previous][Next]>
```

6. Enter 1 or 2 to choose a complete or custom installation respectively.

When you specify a custom installation, the installer enables you to choose the components to install, as shown in the following text:

```

<----- BEA Installer - AquaLogic Data Services Platform 3.0
----->
Choose Components to install:
-----
Release 3.0.0.0
|___ AquaLogic Data Services Platform [1] x
|___ Data Services Server [1.1] x
|___ Data Services Studio [1.2] x
|___ Samples [1.3]

Enter number exactly as it appears in brackets to toggle selection OR
[Exit][Previous][Next]>

```

Enter the value for the component, for example, 1.1 for the ALDSP Runtime. The installer prompts you for additional component values. Continue selecting components, and press Enter without a value when you are done.

7. Enter the Eclipse installation options. Option 1 allows you to install Eclipse bundled with ALDSP. Option 2 allows you to specify a different Eclipse home directory and not install the Eclipse bundle available with ALDSP.
8. The installation begins and the installation status is displayed as it progresses. When the installation is complete, “Congratulations!” is displayed. Press Enter to exit the installer.

This completes the Console Mode installation process.

## Using Silent Console Mode to Install ALDSP

The Windows, UNIX, and Linux versions of the installer provide a noninteractive, or *silent-mode* installation, that you can use in cases when you want to install ALDSP without needing to supply information from the keyboard during the installation process. Instead, the installer gets the required information from a properties file that you provide.

Before launching a silent-mode installation, make sure that all installation prerequisites are met and that all the information in the properties file is correct. After the silent installer is started, it proceeds in the background and does not report exceptions. Some exceptions are ignored.

However, if a previously installed copy of ALDSP is detected, a dialog box will appear asking if you want to override the old version. Other exceptions cause the installer to fail. For example, if the specified BEA home directory <bea\_home> or the specified ALDSP install directory (USER\_INSTALL\_DIR) do not exist or are incorrect, the installer fails.

**Note:** On UNIX systems, the installer displays the message `Installation Complete` when it finishes. This message does not necessarily indicate that the installer was successful; it means only that the process has finished running.

If a fatal exception occurs during installation, the installer displays a message and no changes are made to the system.

## Using Silent Mode on Windows and UNIX Systems

This section describes how to install ALDSP on Windows and UNIX systems using Silent Mode.

To install using silent mode:

1. Create the required installer properties file.

The content of the file is described in “[Exploring the Silent Mode Installer Properties File](#)” on page 3-8. You can use any legal file name for the installer properties file. Verify that the `<bea_home>`, `<USER_INSTALL_DIR>`, and other values specified in the properties file are correct and that all requirements have been met.

2. Open a command window.
3. Navigate to the directory containing the ALDSP installer.

- On Windows, the installer application is:

```
aldsp_300_win32.exe
```

- On UNIX and Linux, the installer application is:

```
filename.bin
```

where `filename.bin` is the name of the ALDSP installation program specific to your platform; for example, `aldsp_300_solaris32.bin` for the Solaris version of ALDSP.

You will find similarly-named installer files for all UNIX and Linux versions of ALDSP.

4. Run the installer application, specifying the properties file and the log file name as options.
- On Windows, run the following command:

```
aldsp_300_win32.exe -mode=silent  
-silent_XML=<drive:\properties_file_path>  
-log=<drive:\log_file_path>
```

where *drive* is the letter that identifies the hard disk drive, *properties\_file\_path* is the complete pathname of the ALDSP silent installation properties file, and *log\_file\_path* is the complete pathname of the log file.

For example, you could enter the following:

```
aldsp_300_win32.exe -mode=silent
-silent_XML=c:\temp\silent.xml
-log=c:\temp\logfile.txt
```

You are returned to the command prompt and the ALDSP installation preparation dialog box is briefly displayed. The installation proceeds in the background with the information specified in the installer properties file.

To verify that the installer is running, open the Windows Task Manager. The installer is listed as the `javaw.exe` process.

- On UNIX, run the following command:

```
sh aldsp_300_solaris32.bin -mode=silent
-silent_XML=complete_properties_file_path
-log=complete_log_file_path
```

where *complete\_properties\_file\_path* is the complete pathname of the properties file and *complete\_log\_file\_path* is the complete pathname of the log file. A complete path is required, even when the file resides in the same directory as the `aldsp_300_solaris32.bin` file.

The message `Preparing to Install` is displayed. After the installer decompresses the required files, the installation proceeds with the information specified in the installer properties file. When the process is successful, `Installation Complete` is displayed.

- On Linux, run the following command:

```
./aldsp_300_linux32.bin -mode=silent
-silent_XML=complete_properties_file_path
-log=complete_log_file_path
```

where *complete\_properties\_file\_path* is the complete pathname of the properties file and *complete\_log\_file\_path* is the complete pathname of the log file. A complete path is required, even when the file resides in the same directory as the `aldsp_300_linux32.bin` file.

The message `Preparing to Install` is displayed. After the installer decompresses the required files, the installation proceeds with the information specified in the installer properties file. When the process is successful, `Installation Complete` is displayed.

## Exploring the Silent Mode Installer Properties File

Table 3-1 describes the required installer properties used with a Silent Mode installation.

**Table 3-1 Installer Properties**

Sample Property Setting	Description
BEAHOME=c:\bea BEAHOME=/bea	<p>BEA Home directory. The specified directory must exist on the system. For full installation options that include ALDSP, WebLogic Platform version 8.1 must be installed in the specified &lt;bea_home&gt; directory.</p> <p>For Windows, specify the absolute path, including the drive. Backslashes must be escaped.</p> <p>For UNIX, specify the absolute path.</p>
USER_INSTALL_DIR=c:\bea USER_INSTALL_DIR=/bea/	<p>Product installation directory.</p> <p>For Windows, specify the absolute path, including the drive. Backslashes must be escaped.</p> <p>For UNIX, specify the absolute path.</p>
WLW_INSTALL_DIR	<ul style="list-style-type: none"> <li>Set the BEA home directory (&lt;bea_home&gt; = c:\bea) where you need to install WebLogic Platform 9.2.</li> </ul>
INSTALL_MERCURY_PROFILING_TOOLS	<ul style="list-style-type: none"> <li>Set this option to true to install this feature.</li> </ul>
INSTALL_NODE_MANAGER_SERVICE	<ul style="list-style-type: none"> <li>Install Node Manager as a Windows service. The default value is set to no and you need Administrative privileges to set this value to yes.</li> </ul>
NODEMGR_PORT	<ul style="list-style-type: none"> <li>Set the Node Manager listen port number. If you do not specify a port, then the installer uses default port 5556. To set this value the INSTALL_NODE_MANAGER_SERVICE option needs to be set to yes.</li> </ul>

**Table 3-1 Installer Properties (Continued)**

Sample Property Setting	Description
COMPONENT_PATHS	<ul style="list-style-type: none"> <li>Specify the components and subcomponents to install.</li> </ul>
INSTALL_SHORTCUTS_IN_ALL_USESRS_FOLDER	<ul style="list-style-type: none"> <li>Shortcuts for all users while installing on Windows. You can set this option to true to install shortcuts for all users or false to install shortcuts for the local user only.</li> </ul> <p><b>Note:</b> You need administrative privileges to install shortcuts in the All Users folder.</p>
ECLIPSE32_HOME	<ul style="list-style-type: none"> <li>Specify the home directory where the Eclipse bundle would be installed.</li> </ul>

The following shows a sample Silent Mode installer properties file (silent.xml):

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
- <!-- Silent installer option: -mode=silent -silent_xml=C:\bea\silent.xml
  -->
- <!--
```

Specify values for each data-value name as follows:

**BEAHOME** Complete pathname to the BEA Home directory that will contain this installation.

**USER\_INSTALL\_DIR** Complete pathname to the product installation directory in which to install the software.

**WLW\_INSTALL\_DIR** Complete pathname for the directory where you want to install BEA Workshop for WebLogic Platform.

## Installing ALDSP Using Console or Silent Mode

### INSTALL\_MERCURY\_PROFILING\_TOOLS

Optionally install Mercury profiling tools.  
Possible values are "true", "yes", "false"  
and "no". The default is "true".

INSTALL\_NODE\_MANAGER\_SERVICE Install Node Manager as a Windows  
service. The default is "no".  
Requires Administrator privileges.

NODEMGR\_PORT Node Manager listen port number. If none specified,  
installer uses default port 5556.  
INSTALL\_NODE\_MANAGER\_SERVICE must be set to "yes"

COMPONENT\_PATHS Components and subcomponents to be installed.  
To install multiple components, separate the components  
with a bar (|).  
To install subcomponents, specify a  
component/subcomponent combination. For example,  
to install Web Server Plug-Ins, use  
"WebLogic Server/Web Server Plug-Ins".

### INSTALL\_SHORTCUT\_IN\_ALL\_USERS\_FOLDER

The Windows Start menu folder in which you want the  
Start menu shortcuts created.

The user performing the installation must have  
Administrator privileges to install the shortcuts

in the All Users folder.

Possible values:

true/yes The shortcuts are created in  
the All Users folder (default)

false/no The shortcuts are created in  
the local user's folder

ECLIPSE32\_HOME

See the Installation Guide for more information.

```
-->
- <bea-installer>
- <input-fields>
  <data-value name="BEAHOME" value="C:/bea922" />
  <data-value name="USER_INSTALL_DIR" value="C:/bea922/aldsp_3.0" />
  <data-value name="COMPONENT_PATHS" value="AquaLogic Data Services
Platform/Data Services Server|AquaLogic Data Services Platform/Data
Services Studio|AquaLogic Data Services Platform/Samples" />
  <data-value name="INSTALL_SHORTCUT_IN_ALL_USERS_FOLDER" value="yes" />
  <data-value name="ECLIPSE32_HOME" value="C:\eclipse" />
</input-fields>
</bea-installer>
```

## Installing ALDSP Using Console or Silent Mode

# Post-Installation Tasks

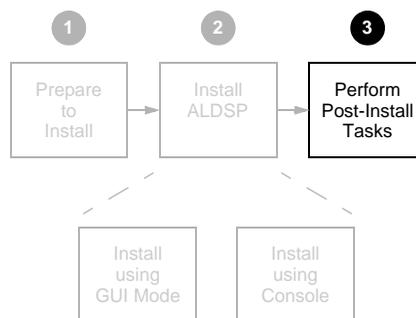
This chapter describes tasks you can do after installation of an instance of AquaLogic Data Services Platform, including verifying the installation, starting development of applications, and exploring Windows shortcuts and UNIX paths. The chapter also describes how to uninstall ALDSP, if required.

The chapter includes the following sections:

- [Verifying the Installation](#)
- [Starting the Development of a Data Integration Solution](#)

Figure 4-1 illustrates the current step in the overall installation process.

**Figure 4-1 Post-Installation Step**



## Verifying the Installation

**Note:** If you are using ALDSP 3.01 you can skip this section. Instead verify a successful installation by following the steps described in:

[How To Configure the Retail Dataspace Sample Application for ALDSP 3.01](#)

To verify that the ALDSP 3.0 installation is successful:

[Step 1: Start Data Services Studio](#)

[Step 2: Configure the ALDSP-enabled Server Environment](#)

[Step 3: Start the Server](#)

### Step 1: Start Data Services Studio

You can use either Workshop for WebLogic 9.2 or Eclipse to run Data Services Studio. In this section, Eclipse 3.2.2 has been used.

1. Open the Data Services Studio using the following Windows Start menu command:  
Start > All Programs > BEA Products > AquaLogic Data Services Platform 3.0 > Data Services Studio
2. From the Workspace Launcher dialog box, select a workspace for ALDSP dataspace projects. By default, the workspace location is as follows:  
`<bea_home>\aldsp_3.0\samples\workspaces\aldsp`
3. Click OK.
4. If this is the first time you have opened Data Services Studio, the ALDSP Welcome screen appears.
5. In the Install Sample Application section click Retail Dataspace Sample. This displays the Retail Dataspace Sample dialog.

### Step 2: Configure the ALDSP-enabled Server Environment

Some simple domain server configuration is required. These steps are described in the following table.

**Table 4-1 Configuring Actions**

<b>Step</b>	<b>Dialog</b>	<b>Field</b>	<b>Action</b>	<b>Description</b>
1.	Retail Database Sample	Server runtime	Click Installed Runtimes...	
2.	Installed Server Runtime Environments		Click Add	
3.	New Server Runtime	Type of runtime:	Select BEA WebLogic Server 9.2	
4.			Click Next.	
5.	Define a WebLogic Runtime	WebLogic Home	Click Browse	
6.	Browse For Folder	Folder	Locate and select the WebLogic home directory. For example, <bea_home>\weblogic 92	Where <bea_home> is c:\bea
7.			Click OK	
8.	Define a WebLogic Runtime		Click Finish.	
9.	Installed Server Runtime Environments		Click OK.	
10.	Retail Dataspace Sample - Server Configuration	Domain home	Click Browse.	
11.	Browse for Folder	Select domain home.	Locate and click on the ALDSP Sample domain: <aldsp_home>\samples\domains\aldsp	where: <aldsp_home> is c:\bea\aldsp_3.0
12.			Click OK.	
13.	Retail Dataspace Sample		Click Finish	Workspace is built.

Now the WebLogic Server 9.2 is configured for ALDSP and can be started.

## Step 3: Start the Server

An ALDSP-enabled server is a version of WebLogic Server with additional functionality to support ALDSP deployment and runtime. The ALDSP server must be running in order to access sample data and to deploy your project.

To start your server from Data Services Studio:

1. Locate the Servers window. If it isn't visible, use the following option command:  
Window > Show View > Servers
2. In the Server window locate BEA WebLogic Server v 9.2@localhost.
3. Right-click the server name and select Start. The start-up operation may take several minutes.
4. After your project has automatically deployed click OK.

## Starting the Development of a Data Integration Solution

Developing your own data integration solution consists of a design phase and an implementation phase.

- When you are ready to start setting up and configuring your own data sources, refer to the AquaLogic Data Services Platform [Administration Guide](#):
- When you are ready to start mapping source and target XML schemas and constructing queries using WebLogic Workshop, refer to [Data Services Developer's Guide](#):
- When your application developer is ready to access data services, see the [Client Application Developer's Guide](#):

# Uninstalling ALDSP

For most platforms you can automatically remove system-installed, unmodified ALDSP artifacts from your system. Once ALDSP has been uninstalled, you can manually remove any extra files, such as modified samples or other user-created files, which were not removed automatically by the uninstaller.

You should undeploy any ALDSP applications before uninstalling ALDSP from your system. For information about undeploying, see the topic [“Undeploying Deployed Applications”](#) in the WebLogic Server documentation.

## Automated Uninstallation

To uninstall ALDSP:

1. Back up any data you want to retain for use with a new installation.
2. Shut down your server.
3. Run the ALDSP uninstaller.
  - On Windows systems, choose the following:  

```
Start → All Programs → BEA Products → BEA AquaLogic Data Services Platform 3.0 > Uninstall BEA AquaLogic Data Services Platform 3.0
```
  - On UNIX systems, run the following command in cases when you installed using GUI mode (as described in [Chapter 2, “Installing ALDSP Using GUI Mode”](#)):  

```
<bea_home>/aldsp_3.0/uninstall/uninstall.sh
```

Alternatively, run the following command with the `-i CONSOLE` option in cases when you installed using Console mode (as described in [Chapter 3, “Installing ALDSP Using Console or Silent Mode”](#))

```
<bea_home>/aldsp_3.0/uninstall/uninstall.sh -i CONSOLE
```

- On Linux systems a manual uninstallation is necessary. Follow the steps described in [“Manual Uninstallation” on page 5-2](#).
4. After the uninstallation, delete the `aldsp_3.0` folder in the `<bea_home>` directory. In addition, you may need to remove some other files manually. When the uninstall is completed, you may be notified that some files could not be removed. This could be a result of new files that were either generated or user-created after ALDSP was installed.

For example, if you added new target schemas, stored queries, Web services, and so on to the ALDSP repository make sure that you save them elsewhere before proceeding to the next step.

**Note:** The Uninstall option does not remove any ALDSP dataspace or other Workshop applications that you have created if they are created outside the `<aldsp_home>` directory, which is usually `<bea_home>\aldsp_3.0`. It is advised that you created your domains and other projects outside `<aldsp_home>` directory.

## Manual Uninstallation

An automatic installation will not remove all AquaLogic Data Services Platform artifacts, particularly those which have been modified. You can follow the steps in this section to manually remove the product.

**Note:** For some platforms such as Linux, it is necessary to perform a manual uninstallation since no automated uninstallation is provided.

To manually remove an installation of AquaLogic Data Services Platform follow these steps:

1. Back-up any data you want to retain for use with a new installation.
2. If running, shut down your server.
3. Delete all the ALDSP binaries and other artifacts found under `<bea_home>`. These directories are listed in [Table 5-1](#).

**Table 5-1 Directories Requiring Manual Removal Following an Uninstall**

Directory	Location
-----------	----------

**Table 5-1 Directories Requiring Manual Removal Following an Uninstall**

<b>ALDSP home directory</b>	<code>&lt;bea_home&gt;\aldsp_3.0</code> <code>&lt;bea_home&gt;/aldsp_3.0</code>
<b>ALDSP samples directory</b>	<code>&lt;bea_home&gt;\aldsp_3.0\samples\</code> <code>&lt;bea_home&gt;/aldsp_3.0/samples/</code>
<b>ALDSP Eclipse installation directory</b>	Default location when installed with ALDSP: <code>&lt;bea_home&gt;\tools\eclipse_pkgs</code>  <b>Note:</b> If Eclipse 3.2.2 has been installed separately (not under <code>&lt;bea_home&gt;\tools\eclipse_pkgs</code> ) and was being used for ALDSP, then you need manually clean up the eclipse configuration directory to avoid known issues in Eclipse 3.2.2. All files in Eclipse WTP home's configuration directory should be deleted except <code>config.ini</code> and files under <code>.settings</code> subdirectory.

4. Open the following file with a text editor:

```
<bea_home>/registry.xml
```

5. Locate the entry for ALDSP. Here is an example:

```
- <product format="1.0" name="AquaLogic Data Services Platform">
- <release level="3.0" ServicePackLevel="0" PatchLevel="0"
  Status="installed" BeaProgramGroup="BEA Products (BEAHOME 2)"
  StartMenu="C:\Documents and Settings\All Users\Start Menu\Programs"
  InstallTime="Nov 13, 2007 5:39:37 PM" InstallDir="C:\bea\aldsp_3.0"
  JavaHome="C:\bea\jrocket90_150_10" JavaVersion="1.5.0_10"
  JavaVendor="BEA">
- <component name="AquaLogic Data Services Platform" version="3.0.0.0">
  <component name="Data Services Server" />
  <component name="Data Services Studio" />
  <component name="Samples" />
</component>
</release>
</product>
```

## Uninstalling ALDSP

6. Remove the entire entry.
7. Proceed with a new installation and deployment.

# Upgrading from ALDSP 2.5 to ALDSP 3.0

ALDSP 2.5 applications are created in the WebLogic Workshop 8.x environment and deployed on WebLogic Server 8.1. ALDSP 3.0 applications are created within the Eclipse framework and are deployed on WebLogic Server 9.2. While upgrading an ALDSP 2.5 application to 3.0, the data service projects within the data service application are upgraded to single a dataspace project. The dataspace project contains all the data services along with the relevant artifacts, which need to be deployed on WebLogic Server 9.2.

This chapter provides the steps to upgrade ALDSP 2.5 applications to ALDSP 3.0 dataspaces. It includes the following sections:

- [Upgrading from ALDSP 2.5 Prerequisites](#)
- [Upgrading ALDSP 2.5 Artifacts to ALDSP 3.0](#)
- [Upgrading an ALDSP Control from ALDSP 2.5 to 3.0](#)
- [Importing ALDSP 2.5 Configuration to 3.0](#)
- [Post-Upgrade Artifact Mapping in ALDSP 3.0](#)
- [ALDSP 2.5 Upgrade: Known Issues and Workarounds](#)

## Upgrading from ALDSP 3.0 to 3.01

The upgrade path from ALDSP 3.0 to 3.01 is very simple: uninstall ALDSP 3.0 and install ALDSP 3.01, as described in [Preparing to Install ALDSP](#). You can then use an existing ALDSP workspace, or create a new one and import the contents of a 3.0-level workspace.

## Upgrading from ALDSP 2.5 Prerequisites

Before you start upgrading an ALDSP 2.5 data service application to 3.0, make sure that you perform Build > Clean Application in Workshop 8.1 environment. This ensures that application upgrade is successful and does not require manual clean up of files later.

## Upgrading ALDSP 2.5 Artifacts to ALDSP 3.0

ALDSP source upgrade allows you to upgrade ALDSP 2.5 data service projects and brings them under 3.0 eclipse workspace as a dataspace project. It also upgrades data service related artifacts such as schema, model diagram, .java files, and WSDL files.

**Note:** To upgrade any ALDSP 2.5 data service applications developed in versions prior to 2.5, you need to first upgrade the application to ALDSP 2.5. For more information about upgrading to ALDSP 2.5, refer to [Upgrading AquaLogic Data Services Platform Application To Version 2.5](#) in the *ALDSP 2.5 Installation Guide*.

This section describes the upgrade of the following artifacts from ALDSP 2.5 to ALDSP 3.0:

**Note:** For steps to upgrade an ALDSP Control to 3.0, refer to [Upgrading an ALDSP Control from ALDSP 2.5 to 3.0](#).

- Application
- Projects
- Schemas
- Java Files within the ALDSP Project
- JAR Data Service Projects
- Shredder Based Artifacts
- Other Artifacts

### Notes:

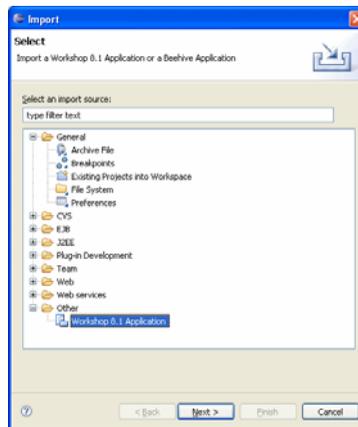
- You cannot use Data Services Studio with Eclipse WTP to upgrade non-data service projects such as Portal and Java web services. To upgrade non-data service projects, refer to the [Upgrading WebLogic Workshop 8.1 Applications](#) chapter in the *Workshop for WebLogic Platform Programmer's Guide*.
- Java projects and Schema projects on which the data service project depends, can be upgraded along with data service projects.

## Upgrading an ALDSP 2.5 Application

To upgrade using the Eclipse IDE:

1. Open the Eclipse IDE.
2. Right-click in the Project Explorer.
3. Select Import. This displays the Import dialog box as shown in [Figure 6-1](#).

**Figure 6-1** Import Dialog Box



4. Expand the Other option and select Workshop 8.1 Application.
5. Click Next. The Workshop 8.1 Application Upgrade: Application Import dialog box is displayed.
6. Browse and select the 2.5 application to be upgraded as shown in [Figure 6-2](#).

**Note:** All projects within a 2.5 application along with dependent schemas and Java projects are upgraded to the same dataspace in 3.0. Select all the Data Service projects dependents such as Java projects and Schema projects if any. If you do not want to upgrade all the projects within the application to the same dataspace, then select only the projects that you want to upgrade.

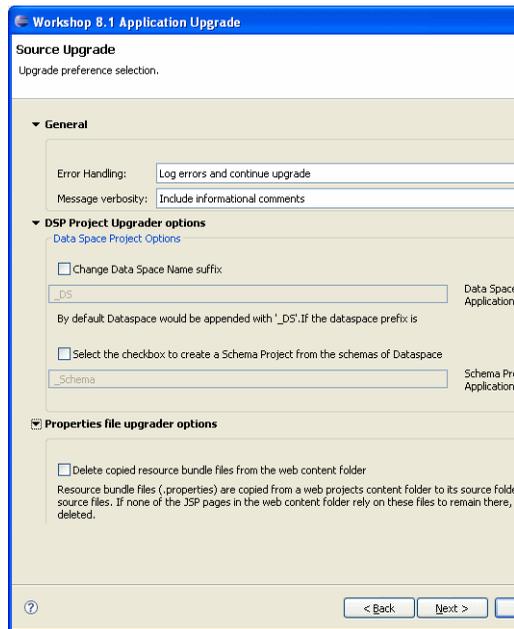
Figure 6-2 Workshop 8.1 Application Upgrade Dialog Box



7. Select the Add this prefix to project names option if you want to specify a prefix for the project being upgraded.

**Note:** All project files upgraded from ALDSP 2.5 will include the prefix including java and schema projects created from data service projects. Deployment name property also includes the prefix. However, the ALDSP 2.5 data service project folder within the dataspace name will not carry the prefix and therefore, the LD namespace URI does not need to be changed.

8. Ensure that runtime environment is set to BEA WebLogic Server v9.2.
9. Click Next. The Workshop 8.1 Application Upgrade: Source Upgrade dialog box is displayed as shown in [Figure 6-3](#).

**Figure 6-3 Workshop 8.1 Application Upgrade: Source Upgrade Dialog Box**

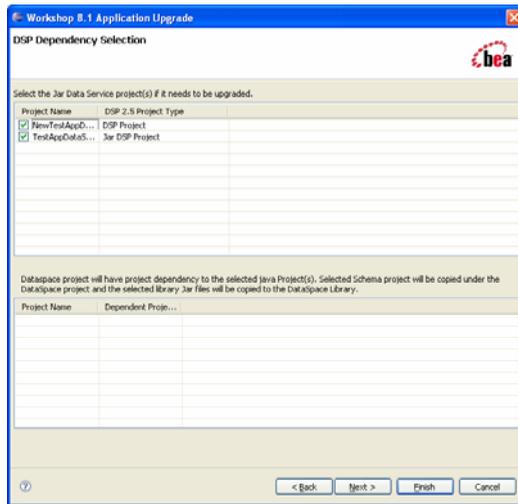
10. In the Source Upgrade dialog box:

- a. Select the Error Handling option, which include:
  - Log errors and continue upgrade
  - Log errors and abort upgrade
  - Display dialog on error
- b. Select the Message verbosity option from the list. The options include:
  - Include informational comments
  - Include warning comments
  - Include error comments
- c. Select the Change Dataspace Name suffix option to change the name of the dataspace suffix. By default, the dataspace project name will be <application\_name>\_DS and can be changed. You can create a schema project by selecting create a schema project from the schemas of Dataspace. This option creates a schema project with all the schemas within

the dataspace project with the name <Application\_Name>\_Schema. When the schema or the Xmlbeans created from these schema is used outside the project then use this option to create a separate schema project.

11. Click Next. This displays the Workshop 8.1 Application Upgrade: DSP Dependency Selection dialog box.

**Figure 6-4 Workshop 8.1 Application Upgrade: DSP Dependency Selection Dialog Box**



This dialog box lists the data service project selected for upgrade along with project JARs including catalog services project (if available). By default, the data service JAR projects are selected for upgrade. You need to review and deselect them if you do not want them to be upgraded. Because the data service projects are already selected the user will not be able to edit the selection here.

It also lists the dataspace dependencies such schemas, Java projects, and library jars. You need to select the Java projects, Schema projects or any library JARs if they are used by one or more data service projects that are getting upgraded. However, internal jar files, `ldclient.jar` and `ld-server-app.jar` are not listed.

12. Click Next. This completes the upgrade process and a summary of the upgraded files is displayed in the Upgrade Preview as shown in [Figure 6-5](#).

Figure 6-5 Upgrade Preview

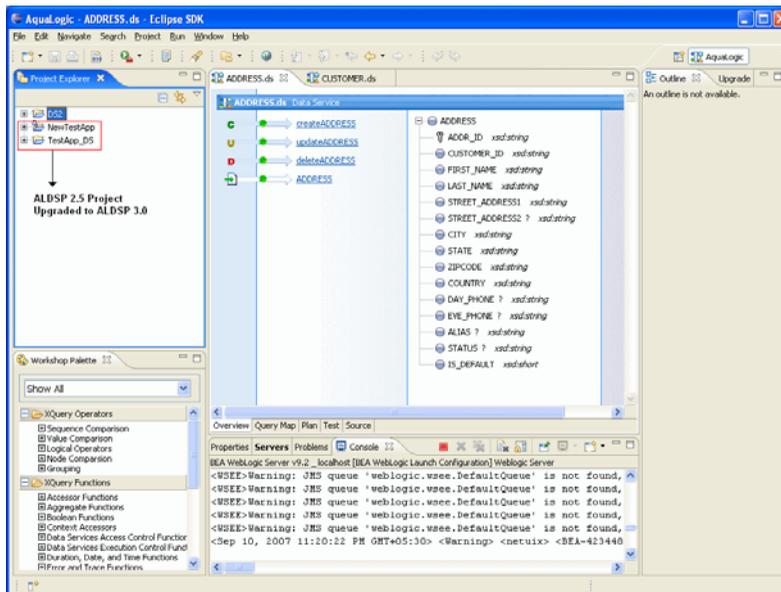


The Upgrade Preview provides the following details:

- List of the actions performed during the upgrade
- List of files added and deleted
- List of files that do not require an upgrade

13. Click Finish. The imported dataspace shows up in the Project Explorer as shown in [Figure 6-6](#).

Figure 6-6 Imported Dataspace in the Eclipse IDE



**Note:** After completing upgrade, an empty EAR project folder is also created. This EAR project is not relevant to the upgraded dataspace and can be deleted if you are not planning to use it at a later stage.

## Upgrading an ALDSP Control from ALDSP 2.5 to 3.0

An ALDSP 2.5 control in Workshop for WebLogic 8.x needs to be upgraded to Workshop for WebLogic 9.2. This section describes the steps to upgrade projects with 2.5 LD control (.jcx).

After the upgrade, all the old annotations are also upgraded to the new annotations. The upgraded ALDSP control is available in Workshop for WebLogic 9.2.

Before you begin the upgrade process, the control plug-in and the control upgrade plug-in needs to be installed:

1. Copy the `com.bea.dsp.ide.control.feature.link` file from the following location:

```
<bea_home>\weblogic92\aldsp_3.0\eclipse-plugins\workshop9
```

where `<bea_home>` is the home directory for BEA product installations. For example, `c:\bea`.

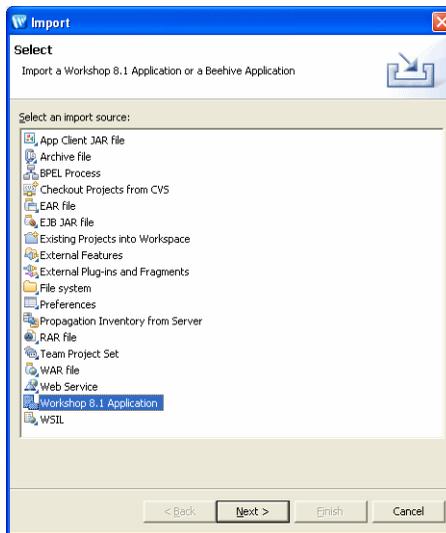
2. Paste this file to the following location:

<bea\_home>\workshop92\eclipse\links

To upgrade the control:

1. Launch Workshop for WebLogic 9.2.
2. Right-click the Package Explorer and select Import.
3. From the Import dialog box, select Workshop 8.1 Application option as shown in [Figure 6-7](#).

**Figure 6-7 Import Dialog Box: Selecting Workshop 8.1 Application Option**

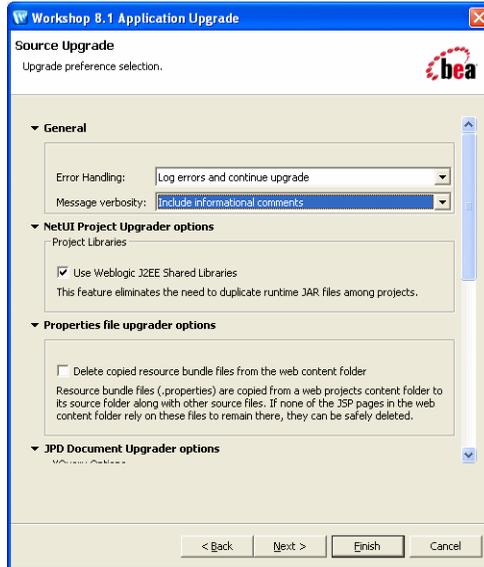


4. Click Next. The Workshop 8.1 Application Upgrade: Application Import dialog box is displayed as shown in [Figure 6-8](#).

**Figure 6-8 Workshop 8.1 Application Upgrade: Application Import Dialog Box**



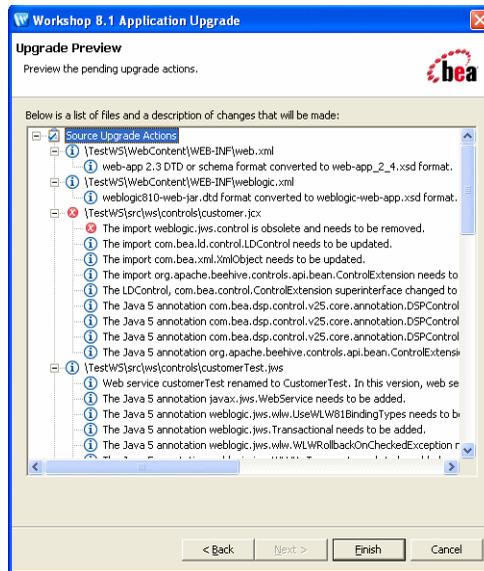
5. In the Application Import box:
  - a. Browse and enter the application that needs to be upgraded.
  - b. If you want to enter a prefix for projects, select the Add this prefix to project names option and enter the prefix.
  - c. Click Next. The Workshop 8.1 Application Upgrade: Source Upgrade dialog box is displayed as shown in [Figure 6-9](#).

**Figure 6-9 Workshop 8.1 Application Upgrade: Source Upgrade Dialog Box**

6. In the Source Upgrade dialog box:
  - a. Select the Error Handling option. The options are same as mentioned in [step a](#) of [Upgrading an ALDSP 2.5 Application](#).
  - b. Select the Message verbosity option. The options are same as mentioned in [step b](#) of [Upgrading an ALDSP 2.5 Application](#).
  - c. Select the Use WebLogic J2EE Shared Libraries option to avoid duplicating runtime JAR files among projects.
  - d. Select the option: Delete copied resource bundle files from the web content folder, to delete the `.properties` files that are not required by the JSP pages.
  - e. Select the Upgrade XQ2002 to XQ2004 option from the JPD Document Upgrader options section to upgrade XQuery from XQ2002 to XQ2004.
  - f. Select the Replace BEA NetUI tag with Apache Beehive tags option to upgrade the NetUI compatibility tags, which are compatible with 9.2 server runtime, with the Apache Beehive project tags.
  - g. Click Next. This starts the process of importing configuration for the project folders and report generation of the files being upgraded.

After the report is generated, the Upgrade Preview dialog box is displayed as shown in [Figure 6-10](#).

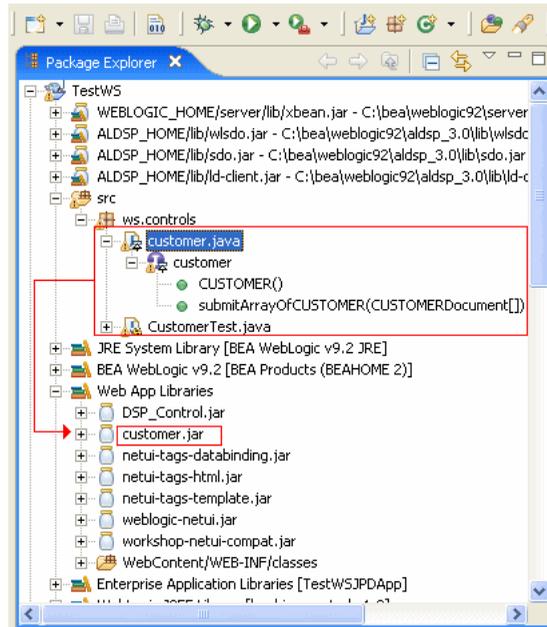
**Figure 6-10 Upgrade Preview**



The control `customer.jcx` is listed and the actions to be performed by the upgrade utility on the associated files are mentioned.

7. Click Finish. This completes the upgrade process. The new control in Workshop for WebLogic 9.2 is displayed in the Project Explorer, as shown in [Figure 6-11](#). After the upgrade, the control is converted to a `.java` file.

Figure 6-11 Upgraded ALDSP Control and Associated Schema in Project Explorer



**Note:** If the return type of the control is complex, then make sure that the upgraded control has the associated schema JAR or XMLBeans JAR in the `WebContent\WEB-INF\lib` folder. For details on XMLBeans JAR file for the control, refer to [Key Differences for WebLogic Workshop 8.1 Users](#) topic in the WebLogic Workshop 9.2 documentation.

## Importing ALDSP 2.5 Configuration to 3.0

After you upgrade an ALDSP 2.5 application to 3.0, you may need to import the old administration configuration to the new ALDSP environment. If you want to use the ALDSP 2.5 project configurations, then you can import the 2.5 configuration file using the ALDSP Administration Console in ALDSP 3.0. The ALDSP 2.5 configuration file is located at:

```
<domain_dir>\liquiddata\<>applicationname>config.xml
```

where `<domain_dir>` is usually located at `<bea_home>\weblogic81\samples\domains\`

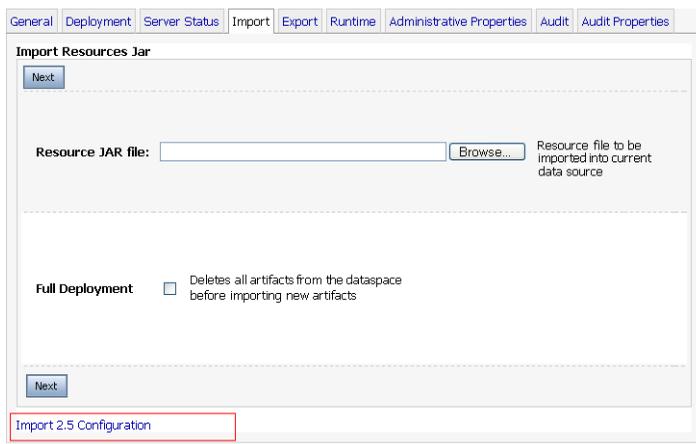
**Note:** The Allow Default Anonymous Access option from ALDSP 2.5 is not imported into 3.0. For more information about setting default anonymous access in ALDSP 3.0, refer to “Allowing Anonymous Access” section in [Securing ALDSP Resources](#).

ALDSP configuration import requires the 2.5 data service projects to be upgraded to a 3.0 dataspace project and this needs to be deployed in the ALDSP 3.0 runtime environment.

To import the 2.5 configurations:

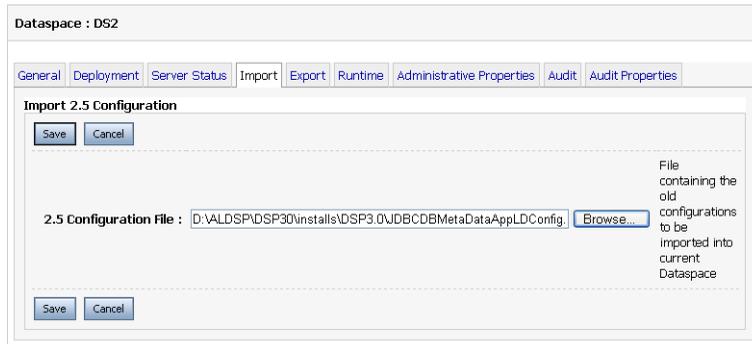
1. Launch ALDSP Administration Console using the following URL and then log in:  
`http://localhost:7001/dspconsole`
2. Select the upgraded dataspace project.
3. Select the System Administration category and then click the Import tab.
4. Click Lock & Edit to acquire the lock.
5. Click the Import 2.5 Configuration link as shown in [Figure 6-12](#).

**Figure 6-12 Import Tab**



6. In the Import 2.5 Configuration section, browse and specify the path where the 2.5 configuration file is located, as shown in [Figure 6-13](#).

Figure 6-13 Importing 2.5 Configuration



7. Click Save > Activate Changes to complete upgrading the configuration file.

## Post-Upgrade Artifact Mapping in ALDSP 3.0

This section highlights the end results after upgrading from ALDSP 2.5 to 3.0 and provides information about mapping artifacts between 2.5 to 3.0.

**Note:** After upgrade, you can continue to use the 2.5 static clients to communicate with the 3.0 server. If you want to change the upgraded dataspace to use the 2.5 clients, then you need to apply the changes to the original 2.5 application, generate the 2.5 static clients and use the clients in ALDSP 3.0.

[Table 6-1](#) provides details about the upgrade and the mappings of artifacts after upgrading to ALDSP 3.0.

**Table 6-1 Upgrade Features and Artifact Mapping**

2.5 Artifact	3.0 Mapping	Features
Application	Dataspaces	<p>The dataspaces name is the 2.5 application name with the <code>_DS</code> suffix. For example, <code>RTLApp_DS</code>. You can change the suffix during the upgrade process. For more information, refer to <a href="#">step c</a> in <a href="#">Upgrading an ALDSP 2.5 Application</a>.</p>
Projects	Dataspaces projects under the upgraded dataspaces.	<ul style="list-style-type: none"> <li>Multiple data service projects are upgraded to a single dataspaces. The data service project becomes a folder under the dataspaces.</li> <li>The project retains the URI for data service and schemas.</li> <li>To allow ALDSP 2.5 mediator APIs access to the upgraded application, the dataspaces project uses the application name (EAR project name) for deployment.</li> </ul>
Project Dependencies	JAR files, schema projects, and Java projects	<ul style="list-style-type: none"> <li>Schema projects that you mark as dependent projects are copied under the dataspaces project. For information about marking or selecting project dependencies, refer to <a href="#">step 11</a> in <a href="#">Upgrading an ALDSP 2.5 Application</a>.</li> <li>Library files marked as dependent are copied to the dataspaces project's <code>DSP-INF\lib</code> folder.</li> <li>Dataspaces project depends on the created Java projects and dependant Java projects. All the created Java projects and dependant Java projects depends on the schema project, if created.</li> </ul>

**Table 6-1 Upgrade Features and Artifact Mapping**

2.5 Artifact	3.0 Mapping	Features
Other Project Files	index.xml	<ul style="list-style-type: none"> <li>Individual index.xml files are removed from projects within the upgraded dataspace. Instead, a single index.xml file is created for the entire dataspace when you build the project.</li> <li>For different projects within a dataspace, the sdo.xsdconfig.xml file is removed at the root level.</li> <li>The xquery-types.xsd file is removed for individual projects and one file is created for the dataspace.</li> <li>The SQL index file from each of the projects in a dataspace are merged and placed under DSP-INF/sql/sql-index.xml for the dataspace.</li> </ul>
Java Files	_java Suffix	<ul style="list-style-type: none"> <li>In 3.0, dataspace projects cannot store Java files. Therefore, during upgrade of 2.5 data service projects, they are moved under a new Java project for each data service project. This project has the name &lt;dataservice_project_name&gt;_java</li> <li>For the Java projects that are created from the data service project, the dependency is set at the dataspace level.</li> <li>While building the dataspace project, the dependent JAR or binary files, are copied to the DSP-INF/lib folder.</li> </ul>

**Table 6-1 Upgrade Features and Artifact Mapping**

2.5 Artifact	3.0 Mapping	Features
Schema Files	Same as 2.5	<p>The folder structure and name of the schema remains the same after upgrade.</p> <p>If you choose to create a separate schema project, then all the schema files from different selected data service projects are placed in a separate schema project with name <code>&lt;dataspace_name&gt;_Schema</code>. For information about selecting separate schema project, refer to <a href="#">step c in Upgrading an ALDSP 2.5 Application</a>.</p>
Catalog Data Service Projects	Exploded as <code>_catalogservices</code> folder under <code>dataspace</code> project	<p>Catalog services JAR under <code>APP-INF/lib</code> in ALDSP 2.5 are upgraded in the exploded form under the <code>dataspace</code>.</p> <ul style="list-style-type: none"> <li>• All catalog services are upgraded like other data services.</li> <li>• All class files are filtered out and deleted after upgraded and a new <code>catalogservicelib.jar</code> is added under <code>DSP-INF/lib</code> folder.</li> </ul>

There are other artifacts such as data services, xfl and Java files, schemas, and WSDL files, which are mentioned in [Table 6-2](#).

**Table 6-2 Mapping and Features of Other Artifacts Post-Upgrade**

Other Files	Features
.ds	<ul style="list-style-type: none"> <li>• The .ds files are upgraded as entity data services with the .ds extension.</li> <li>• Data service namespaces are not changed.</li> <li>• Pragma information that contains the version number changes to include the 3.0 stamp.</li> <li>• All read functions are upgraded with <code>visibility=public,kind=retrieve</code></li> <li>• All navigation functions are retained with <code>visibility=public, kind=navigate</code></li> <li>• All procedures (side effecting functions) have <code>visibility=public, kind=library</code>. In addition: <ul style="list-style-type: none"> <li>– Function declarations use the following declaration for external functions: <pre style="margin: 0;">procedure declare procedure for external functions</pre> </li> <li>– Function declaration use the following declaration if the function is not external and has a body: <pre style="margin: 0;">function (declare function)</pre> </li> </ul> </li> </ul> <p><b>Note:</b> After the upgrade, mark the kind for side-effecting functions as CUD if they are from the libraryProcedure.</p> <ul style="list-style-type: none"> <li>• All functions are marked as <code>primary = false</code> with the new validation rules.</li> <li>• A key schema file is created and added, if key element is present in the data service pragma. This key schema is <code>TargetTypeSchemaName_KEY</code>. It shares the same namespace as the target type schema such as XML or other return type schemas.</li> </ul>

**Table 6-2 Mapping and Features of Other Artifacts Post-Upgrade**

Other Files	Features
.xfl	<ul style="list-style-type: none"> <li>• These files are changed to library data service files and have no return type pragma.</li> <li>• The namespace prefix are changed from lib to ld and all its references are also modified.</li> <li>• Name of the xfl is retained if there are no DS files with the name in the same folder. If another data service with the same name exists, then it is resolved with numbered suffixes.</li> <li>• All functions are set to <code>visibility=protected</code> during upgrade and in case of private functions, are retained as private.</li> <li>• All functions are set to <code>kind=library</code> and function declaration is <code>function(declare function)</code></li> </ul>
.java files	<ul style="list-style-type: none"> <li>• These files are moved to separate Java projects.</li> </ul>
WSDL files	WSDL files are upgraded as is. However, in some cases, you may need to upgrade the WSDL manually.
XML, CSV	No change occurs to these files after upgrade.
Model Diagram	Model diagrams are upgraded as is.

**Note:** If you have configured the max thread per application property in ALDSP2.5 using the ALDSP Administration Console, then this configuration is not upgraded to 3.0.

## Post-Upgrade Tasks for Java Web Service

After completing the upgrade of Java Web Service (JWS), you need to perform the following steps, to retrieve values from a dynamic JWS client:

1. Save the WSDL file for the JWS locally and remove the `parameterOrder` attribute from operations tag.
2. Modify the client code to use the saved WSDL to create the service. Examples of the ALDSP 2.5 configuration and the modified version are shown in [Listing 6-1](#) and [Listing 6-2](#).

**Listing 6-1 ALDSP 2.5 Client**

---

```

QName serviceName = new QName(targetNamespace, 'testSimplePhyCtrlTest');
URL wsdlLocation = new
URL('http://localhost:7001/testWSProj/controls/testSimplePhyCtrlTest.jws?W
SDL');
Service service = factory.createService(wsdlLocation, serviceName);

```

**Listing 6-2 Modified ALDSP 2.5 Client**

---

```

QName serviceName = new QName(targetNamespace, 'testSimplePhyCtrlTest');
URL wsdlLocation = new
URL('file:///D:/dynwsclient/testSimplePhyCtrlTest.wsdl');
Service service = factory.createService(wsdlLocation, serviceName);

```

**Note:** While modifying the client, ensure that the port name matches the port name of the migrated JWS.

3. Recompile and run the client with the ALDSP 2.5 classpath. This allows you to run the JWS successfully.

## ALDSP 2.5 Upgrade: Known Issues and Workarounds

[Table 6-3](#) are some of the known issues after you upgrade various components from ALDSP 2.5 environment to 3.0:

**Table 6-3 Known Issues and Workarounds**

Known Issues	Workaround
<p>An XQueryTypeException occurs when a read function, which is shredded from a custom java function, is invoked.</p>	<p>The schema (.xsd) file needs to be recompiled using XMLBeans v2, using the following command:</p> <pre data-bbox="619 513 1166 722"> java -classpath \${WL_HOME}/common/lib/javax.xml.stream_ 1.0.0.0.jar;\${WL_HOME}/server/lib/xbean .jar com.bea.xbean.tool.SchemaCompiler *.xsd -d testbin -src testsrc -out &lt;jarfilename&gt;.jar </pre> <p>where &lt;WL_HOME&gt; is root directory, which contains the WebLogic Platform 9.2 installation and is usually located at:</p> <pre data-bbox="619 840 911 866">&lt;bea_home&gt;\weblogic92</pre> <p>&lt;jarfilename&gt; is the name of the schema JAR file.</p>
<p>ALDSP 2.5 update properties are not displayed in the Properties view in Data Service Studio. For example, DS pragma elements are not displayed.</p>	<p>No workaround available.</p>
<p>ALDSP 2.5 control will work if the SDO is exposed over JWS, however upgraded 2.5 applications will not support exposing SDO over JWS.</p>	<p>No workaround available.</p>

**Table 6-3 Known Issues and Workarounds**

Known Issues	Workaround
The max thread per application property in ALDSP 2.5 is not imported into 3.0 configuration.	<p>You can obtain similar results by configuring the dataspace WorkManager. However, the value is not the same because the pool of thread is shared across async/timeout spawned threads and the top-level EJB threads.</p> <p>In addition, because the WorkManager is self-tuning, it is advised that you do not specify a maximum- or minimum-threads constraint unless you have a specific need to do so.</p> <p>For additional information on the WLS 9.2 WrokManager see:</p> <p><a href="http://edocs.bea.com/wls/docs92/config_wls/self_tuned.html">http://edocs.bea.com/wls/docs92/config_wls/self_tuned.html</a></p>
Schema projects that only exist in JAR format in the Libraries folder are not upgraded to create schemas in the dataspace project.	Obtain the original expanded schema project and include it in the application before starting upgrade.

Upgrading from ALDSP 2.5 to ALDSP 3.0