



BEA

WebLogic Server

Installing and Setting up WebLogic Server 5.1

Document 2.0
April 2000

Copyright

Copyright © 2000 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, WebLogic, jDriver, ObjectBroker, TOP END, and TUXEDO are registered trademarks of BEA Systems, Inc. BEA Builder, BEA Connect, BEA Manager, BEA MessageQ, Jolt and M3 are trademarks of BEA Systems, Inc.

All other company names may be trademarks of the respective companies with which they are associated.

Installing and Starting WebLogic Server 5.1

Document Edition	Part Number	Date	Software Version
2.0	860-001001-002	April 2000	WebLogic Server 5.1

Contents

1. Overview of Installing WebLogic Server 5.1

What's new in WebLogic Server, version 5.1	7
Checking your package	8
Hardware Requirements	9
Software Requirements	9
Upgrading from an earlier release of WebLogic Server	9
Important notes.....	9
Steps to take before installation	10
Steps to take after installation	10
Installing WebLogic Server on your platform.....	11

2. Installing using the InstallShield distribution (Windows NT)

Uninstalling a previous release.....	14
Running the InstallShield program.....	14
Next steps	15

3. Installing from a zip archive (UNIX, Windows NT)

Overview	17
Installing on Windows NT using the zip archive.....	17
Upgrading from a previous release.....	18
Installing from the zip archive.....	18
Next steps	21

4. Setting up and Starting WebLogic Server 5.1

Tasks to set up and start WebLogic Server	24
Install a JDK	24
Native vs. Green Threads	25

Hot Spot.....	25
Setting the environment.....	26
Setting the system PATH	26
Setting the classpath	26
Upgrading from a previous release	27
Setting your Java system classpath	28
Setting your WebLogic classpath.....	28
Microsoft SDK for Java (JView)	29
Starting WebLogic Server Statically	30
Cloudscape DBMS	31
Setting up the Java security manager for Java 2.....	31
Modifying the weblogic.policy file for general use	31
Modifying the weblogic.policy file for third party or user-written classes	32
Installing a WebLogic license	33
Evaluation licenses	33
Other licenses	34
Updating a license	34
Upgrading licenses from a previous release.....	35
Starting WebLogic Server on Windows NT.....	36
Start Menu	36
NT Service.....	36
Windows Convenience Programs.....	37
Starting WebLogic Server from the command line	40
Important note regarding WebLogic RMI over IIOP	40
Requirements for Starting WebLogic Server	40
Command line examples	42
JDK 1.1.x example	42
JDK 1.2 (Java 2) example	42
Jview example	43
Additional options:	43
Starting WebLogic Enterprise Connectivity	43
Starting WebLogic Server from the WebLogic Console.....	44
Starting WebLogic Server using scripts	46
Next steps	46

Set up your development environment.....	46
Install JDBC drivers for use with WebLogic Server.....	46
Oracle	47
Informix	47
Microsoft SQL Server	47
Sybase	47
Other Documentation	47

5. Installing WebLogic jDriver for Oracle

Overview	49
Important issues for release 5.1	50
Other JDBC drivers	50
Installation steps	51
Editing an entry to the XML license file.....	53
Setting your path and client libraries.....	54
JDBC 2.0	55
Platform considerations	55
Directory to put in your system PATH	55
Windows NT	55
Solaris.....	56
IBM AIX.....	57
HP-UX 11	57
SGI IRIX.....	58
Siemens MIPS	58
Compaq Tru64 UNIX	58
Note for Microsoft SDK for Java (Jview) users	59
Checking connections to the Oracle database	59
Setting up a connection pool	60
Configuring a connection pool with WebLogic Server	60
Using the connection pool in your application	61
Using IDEs or debuggers with WebLogic jDrivers	62
Next step.....	62

Index



Preface

- Purpose of This Document
- Who Should Read This Document
- How This Document Is Organized
- How to Use This Document
- Documentation Conventions
- Related Documents
- Contact Information

Purpose of This Document

This document describes how to install and start BEA WebLogic Server.

Who Should Read This Document

This document is intended for system administrators, network administrators, evaluators, and developers who are installing WebLogic Server. It assumes a familiarity with your operating system and platform and a familiarity with the Java programming environment.

How This Document Is Organized

Installing and Starting WebLogic Server is organized as follows:

- Chapter 1, “Overview of Installing WebLogic Server 5.1,” provides an overview of installation requirements and procedures.
- Chapter 2, “Installing using the InstallShield distribution (Windows NT),” contains installation procedures for the Windows NT platform, using the InstallShield program. Note that WebLogic Server may also be installed on Windows NT using a zip archive and the instructions in Chapter 3, “Installing from a zip archive (UNIX, Windows NT).”
- Chapter 3, “Installing from a zip archive (UNIX, Windows NT),” contains installation instructions for using the zip archive to install WebLogic Server.
- Chapter 4, “Setting up and Starting WebLogic Server 5.1,” tells you how to configure and start WebLogic Server.
- Chapter 5, “Installing WebLogic jDriver for Oracle,” has procedures for installing the WebLogic type-2 JDBC driver for Oracle.

How to Use This Document

This document, *Installing and Starting WebLogic Server*, is available as a booklet, in HTML or PDF format on the BEA website, and in HTML or PDF format on CD-ROM versions of WebLogic Server 4.5.

To access the online HTML version on the BEA website, open <http://www.weblogic.com/docs50/install/index.html> in your web browser.

If you have WebLogic Server 4.5 on CD-ROM, you can find this book on your CD-ROM at *documentation/docs/install/index.html*. The PDF version can be found at *documentation/pdf/Install_Guide_50.pdf*.

You can view or print the PDF versions of these documents using Adobe Acrobat, version 3.0 or higher.

Complete documentation for WebLogic Server 4.5 is available online at <http://www.weblogic.com/docs51/resources.html> and on CD at <documentation/index.html>.

URLs cited in this document that point to the BEA WebLogic Server website (<http://www.weblogic.com/docs50...>) can also be accessed on the CD by substituting *documentation* for the *www.weblogic.com/docs50* portion of the URL.

Documentation Conventions

The following documentation conventions are used throughout this document.

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

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

Related Documents

The documentation for WebLogic Server 4.5 can be found on the Internet at the URL <http://www.weblogic.com/docs50/index.html>. If you have WebLogic Server 4.5 on CD-ROM, you can view the documentation with a web browser by loading the *documentation/index.html* file on your CD-ROM.

The documentation on the BEA website is updated frequently. You can always find the most current documentation by browsing the BEA documentation website at the above URL.

Contact Information

The following sections provide information about how to obtain support for the documentation and software.

Documentation Support

If you have questions or comments on the documentation, you can contact BEA Documentation Engineering by e-mail at docsupport@beasys.com.

Customer Support

If you have any questions about this version of WebLogic Server, or if you have problems installing and running WebLogic Server, contact Customer Support through BEA WebSupport at <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
 - The name and version of the product you are using
 - A description of the problem and the content of pertinent error messages

1 Overview of Installing WebLogic Server 5.1

This manual contains instructions on installing, setting up, and starting **WebLogic Server** and the **WebLogic jDriver for Oracle** type-2 JDBC driver. A [PDF version](#) of these installation documents is also available.

What's new in WebLogic Server, version 5.1

See the [Release Notes](http://www.weblogic.com/docs51/release_notes/index.html) at http://www.weblogic.com/docs51/release_notes/index.html for complete coverage of new features and upgrade issues.

Users of previous versions of WebLogic Server should take note of the following:

- **Windows NT only:** WebLogic Server is shipped with a Java Runtime Environment (JRE) that contains the Java classes for the Java Development Kit (JDK), version 1.2 (also called Java 2). Previous versions of WebLogic Server were shipped with both a 1.2 and 1.1.7 JRE, selectable from the Windows Start menu or with the **wlconfig** program. These JRE selection methods are no longer available.

You can run WebLogic Server with either version of the JDK, but some WebLogic Server features (such as JDBC 2.0) require the use of a version 1.2 JDK.

- **Windows NT only:** The JRE mentioned above does not include classes that are required for use with certain tools such as the EJB compiler (`ejbc`). To use such tools you must install and configure a JDK for use with Weblogic Server. See

Installing using the InstallShield distribution (Windows NT) on page 13 for more information.

- **Oracle users only:** The WebLogic jDriver for Oracle (formerly called jdbcKona/Oracle) is now available in several versions. The one you uses depends on your Oracle client installation, the version of the Oracle API you will use to connect to the Oracle server, and your platform. You must place the appropriate file in your system's path or shared library path for this driver to function. *Unlike previous releases of WebLogic Server, which had only one such file that was always present in the default path, you now must specify this path setting explicitly.* For details and instructions on setting these options, see Setting your path and client libraries on page 54.
- **WebLogic RMI over IIOP.** If you are using WebLogic RMI over IIOP, you must use JDK 1.3. There are some important restrictions regarding the use of JDK 1.3. For details, see Using WebLogic RMI over IIOP at http://www.weblogic.com/docs51/classdocs/API_rmi_iiop.html#StartServer.

Checking your package

When you open your BEA WebLogic Server box, you will find the following:

- A CD containing the WebLogic Server software and documentation.
- Two manuals:
 - *Installing and Starting WebLogic Server*
 - *Introduction to WebLogic Server*
- Release Notes
- A Software warranty and license description

Hardware Requirements

- 64 MB of RAM
- At least 80 MB of free hard disk space
- One of the certified WebLogic Server platforms. See <http://www.weblogic.com/docs51/platforms/index.html> for more information.

Software Requirements

The software requirements vary among platforms and operating systems and are frequently updated by BEA. Check the Platforms Support page (<http://www.weblogic.com/docs51/platforms/index.html>) on the BEA web site for the latest information on your platform and operating system.

Upgrading from an earlier release of WebLogic Server

Important notes

- **Version 5.1 of WebLogic Server requires a different classpath and command line to start WebLogic Server than was required in earlier releases.** These changes are discussed in detail in this document.
- **You will not be able to use any scripts or shortcuts you may have used to start previous versions of WebLogic Server with this release.** New scripts are included with this release that you can modify for your environment. For more information, see Starting WebLogic Server using scripts on page 46.

- If you are using either the WebLogic jDriver for Informix4 (formerly called jdbcKona/Informix4) or jDriver for Microsoft SQL Server (formerly called jdbcKona/MSSQLServer4) type 4 JDBC drivers, you must also upgrade them for use with WebLogic Server 5.1. You can obtain these drivers from the same location where you obtained WebLogic Server.

Steps to take before installation

1. Save your current license files. Open the directory where your registered installation is located and copy your `webLogicLicense.xml` or `webLogicLicense.class` file into a safe place. You will be instructed where to place your license file later in these instructions.
2. Save your `weblogic.properties` and `weblogic.policy` files to a safe location.
3. Save any user-written code or compiled classes to a safe location.
4. To safeguard your applications and environment, BEA recommends that you copy your entire WebLogic distribution to a safe location or otherwise back up your previous installation. Do not install a new version of WebLogic Server on top of a previous version

Steps to take after installation

1. Re-run the WebLogic utility `ejbc` on your Enterprise Java Beans.
2. Re-run the WebLogic RMI compiler, `rmic` on any existing code to regenerate the wrapper classes so that they are compatible with the new version of WebLogic Server.

Installing WebLogic Server on your platform

The following documents contain detailed instructions for installing WebLogic Server:

- [Installing using the InstallShield distribution \(Windows NT\)](#), on page 13
- [Installing from a zip archive \(UNIX, Windows NT\)](#), on page 17

2 Installing using the InstallShield distribution (Windows NT)

The InstallShield distribution makes it easy to install WebLogic Server on your Windows NT computer. The InstallShield program unpacks the distribution, sets up your WebLogic properties and security policy files, installs a Java runtime environment (JRE), and sets up shortcuts to run WebLogic Server.

You can also install WebLogic Server from a zip archive. See [Installing from a zip archive \(UNIX, Windows NT\)](#) on page 17 for more information. However, using the zip archive does not automatically set up properties, shortcuts, and Windows Registry settings. BEA recommends that Windows NT users use the InstallShield distribution.

If you are upgrading from a previous release of WebLogic Server, you can either uninstall the previous version or install the new version in a different directory. If you are re-installing the same version of WebLogic Server, uninstall it first.

Uninstalling a previous release

To uninstall a previous release:

1. Click **Start**—>**Settings**—>**Control Panel**
2. Double-click **Add/Remove Programs**.
3. Select **WebLogic version x.x.x** (Where **x.x.x** is the WebLogic Server version number.)
4. Click **Add/Remove**.

Running the InstallShield program

1. Download the WebLogic InstallShield distribution, or locate the **weblogic510.exe** file on your CD.
2. Double click the **weblogic510.exe** file. The InstallShield program begins to install WebLogic Server.
3. Follow the instructions on the screen. The InstallShield program prompts you for the following additional information:
 - The name of the directory where you want to install WebLogic Server. If you are upgrading from a previous release, install WebLogic Server into a new directory. BEA recommends that you install WebLogic Server in a top-level directory. For example, `c:\weblogic`.
 - A **System Password**. Select a password that you will use to access administrative functions of the WebLogic Server. If you forget the password, you can retrieve it by looking in the `weblogic.properties` file, under the property `weblogic.password.system`.
 - Evaluation password. If you are installing WebLogic Server from an evaluation CD, go to the URL listed on the CD packaging and complete the registration process. After you submit your online registration, you will

receive a password by email. Enter this password when prompted by the InstallShield.

Note: If you have purchased the CD version of WebLogic Server, you will not need this password. You will receive a set of license keys by email. To use these keys, follow the instructions under [Updating a license](#) on page 34 before starting WebLogic Server.

4. Specify JAVA_HOME.

WebLogic Server ships with a Java Runtime Environment (JRE). This JRE does not include classes that are required for use with certain tools such as the EJB compiler (`ejbc`). To use such tools you must install and configure a JDK for use with Weblogic Server.

If you will be starting WebLogic Server using the Windows **wlserver** native program, from the Windows Start menu, or when running WebLogic Server as a Windows NT Service, you must use the **wlconfig** program to specify the directory containing your Java Development Kit (JDK). For example:

```
wlconfig -JAVA_HOME c:\java
```

For more information, see [Windows Convenience Programs](#) on page 37

Next steps

1. Oracle users only.

If you will be using the supplied Oracle JDBC driver (WebLogic jDriver for Oracle, formerly called jdbcKona/Oracle) you must explicitly set your **PATH** to include both the Oracle-supplied libraries and the appropriate BEA-supplied libraries. See [Installing WebLogic jDriver for Oracle](#) on page 49 for details.

2. Starting WebLogic Server

The WebLogic Server distribution is now installed on your computer. To find out how to set up and start WebLogic Server, please see [Setting up and Starting WebLogic Server 5.1](#), on page 23.

3 Installing from a zip archive (UNIX, Windows NT)

Overview

This section tells you how to install WebLogic Server using the zip archive. This zip archive is required for installing WebLogic Server on supported UNIX systems and may also be used for installation on Windows NT.

Note: When installing WebLogic Server on a UNIX system, do not unzip the distribution on a Windows NT machine and then copy that installation to a UNIX machine.

Installing on Windows NT using the zip archive

Installing WebLogic Server on Windows NT using the zip archive instead of the InstallShield means that the following steps will not be performed automatically:

- Windows Convenience Programs (see [page 37](#)) will not be installed.
- Start menu shortcuts will not be set up.
- The `weblogic.properties` file will not be modified.

- The `weblogic.policy` file will not be modified.
- You will be unable to install WebLogic Server as a Windows NT service.

BEA recommends that you use the InstallShield version when installing on Windows NT. For more information, see [Installing using the InstallShield distribution \(Windows NT\)](#) on page 13.

Upgrading from a previous release

If you have a version of WebLogic Server earlier than **5.1.0**, you should either remove your previous installation or re-name its directory. **Do not install the new release on top of an older release.**

Installing from the zip archive

Note: If you are installing on Windows NT, please ignore the steps labeled **(UNIX only)**.

1. Download the WebLogic distribution, or locate the `weblogic510.zip` file on your CD (located in the root directory).
2. **(UNIX only)** Create a UNIX *user* to install WebLogic Server, such as `weblogic` (optional). This allows you to control access to the distribution and to set permissions for your WebLogic distribution. See the documentation for your operating system for details. You may also want to create a *group* that has permissions to access the distribution.
3. **(UNIX only)** Log in with the WebLogic user you created.
4. Select a directory for the installation. BEA recommends that you install WebLogic Server in a directory where you normally install shared applications, such as `/usr/local/weblogic`. If you are upgrading from a previous version of WebLogic Server, rename or delete the root directory of your old installation.

5. Unzip the distribution. You can use the **unzip** utility to un-pack the distribution (or **winzip** on Windows NT), or, if you already have a Java Development Kit (JDK) installed, you can use the Java **jar** utility. (The **jar** utility is not included with the JRE.) A **weblogic** directory will be created in the directory where you execute the jar command. For example,

```
$ jar -xvf path/weblogic510.zip
```

(where **path** is the path to the **weblogic510.zip** file)

unzips the distribution into a directory called **weblogic**, located beneath the current directory.

6. Edit the **weblogic.properties** file, located in the top-level directory where you unpacked the WebLogic Server distribution. The **weblogic.properties** file contains name-value pairs for properties that set the functionality of the WebLogic Server. To run the server with a basic configuration, you must edit several properties:
 - Administrator's password. Edit the property **weblogic.password.system**, adding your own system administrator's password. This user-selected password is used to access administrative functions of WebLogic Server such as the WebLogic Console. The default minimum length for the password is 8 characters and the maximum is 16 characters (this can be changed, see [Setting WebLogic Properties at http://www.weblogic.com/docs51/admindocs/properties.html#usergroup](http://www.weblogic.com/docs51/admindocs/properties.html#usergroup)). For example,

```
weblogic.password.system=myPassword
```
 - Modify paths to your installed directory. The default **weblogic.properties** file uses a path of **/weblogic** as the default location for various files. If you have installed WebLogic Server into a different directory, replace all the **/weblogic** entries with the correct directory.
 - Modify external paths. The **weblogic.properties** file refers to several directories outside of the WebLogic Server installation. Examine all instances of **c:/** in the **weblogic.properties** file and correct these entries as needed.
 - Install a performance pack. If you are installing WebLogic Server on Windows NT, Solaris 2.6/2.7, or AIX 4.3, you can add a native performance pack that uses a platform-optimized (native) socket muxer to improve server performance. To use a performance pack, make sure the following property is

defined in your `weblogic.properties` file (the default `weblogic.properties` file shipped with your distribution already defines this property and sets it to *true*):

```
weblogic.system.nativeIO.enable=true
```

7. If you are using Java 2 (JDK 1.2), modify the `weblogic.policy` file. This is covered in detail under [Setting up the Java security manager for Java 2](#).
8. **(UNIX only)** Set the file permissions (optional). After you unpack the distribution, the permissions for all of the files will be **644** (read/write access for the user, read access for the group and for all). For security reasons, you may want to change these permissions, particularly on sensitive files such as `weblogic.properties`, `weblogic.log`, and `weblogic.policy`. For these sensitive files, BEA recommends that you set the permissions to **640** (no access for “all”).
9. **(UNIX only)** If you will be running WebLogic Server as a web server listening for requests on port 80, please see [Additional steps on UNIX](#) at <http://www.weblogic.com/docs51/admindocs/http.html#unixsteps>.
10. **(UNIX only)** If you will be using performance packs or WebLogic ZAC, you must set your **load** (or **shared**) **library path** to point to the directory containing the native libraries for your platform. (On most UNIX systems, this is called the `LD_LIBRARY_PATH`, on HP-UX, it is called `SHLIB_PATH`, and on IBM AIX it is called `LIBPATH`. See your operating system documentation for instructions on setting this variable.) These libraries are located in the directory shown in the table below.

Platform	Directory containing shared library
IBM AIX	<code>weblogic/lib/aix</code>
HP-UX 11	<code>weblogic/lib/hpux11</code>
SGI IRIX	<code>weblogic/lib/irix</code>
Red Hat Linux	<code>weblogic/lib/linux</code>
Sun Solaris	<code>weblogic/lib/solaris</code>
Compaq Tru64 Unix	<code>weblogic/tru64unix</code>

(Where *weblogic* is the directory containing your WebLogic Server installation.)

11. (UNIX only) If you are using the HP-UX platform, please see the note on [shared libraries](http://www.weblogic.com/docs51/techsupport/sharedlibs.html#hpux) at <http://www.weblogic.com/docs51/techsupport/sharedlibs.html#hpux>.

Next steps

1. Oracle users only.

If you will be using the supplied Oracle JDBC driver (WebLogic jDriver for Oracle, formerly called jdbcKona/Oracle) you must explicitly set your **SHARED LIBRARY PATH** to include the appropriate BEA-supplied libraries. See [Installing WebLogic jDriver for Oracle](#) on page 49 for details. Your Oracle-supplied client libraries must also be included in the shared library path.

2. Starting WebLogic Server

The WebLogic Server distribution is now installed on your computer. To find out how to set up and start WebLogic Server, please see [Setting up and Starting WebLogic Server 5.1](#), on page 23.

4 Setting up and Starting WebLogic Server 5.1

This section guides you through setting up your computer to run a single WebLogic Server. The steps listed here will prepare your computer to run WebLogic Server in a minimal, basic configuration. As you begin developing and deploying applications using WebLogic Server, you will define additional properties that set up the WebLogic services or Application Program Interfaces (API) that you use in your code. Additional information is available in the [WebLogic Server Developers Guides](http://www.weblogic.com/docs51/classdocs/index.html) for each API at <http://www.weblogic.com/docs51/classdocs/index.html>.

There are several ways you can start WebLogic Server. Window NT users can start WebLogic Server immediately after running the Install Shield by using the Windows [Start Menu](#) (see page 36). The InstallShield distribution also allows Windows users to use [Windows Convenience Programs](#) (see page 37) to start WebLogic Server.

Additional ways of starting WebLogic Server, including [Starting WebLogic Server from the command line](#) (see page 40), and [Starting WebLogic Server using scripts](#) (see page 46) are also described in this document. Follow the tasks listed below to correctly set your environment before using the command line or scripts to start WebLogic Server.

For information on setting up a WebLogic Cluster, please see [Setting up a WebLogic Cluster](http://www.weblogic.com/docs51/cluster/index.html), at <http://www.weblogic.com/docs51/cluster/index.html>, in the WebLogic Server documentation.

Tasks to set up and start WebLogic Server

1. Install a JDK (see page 24)
2. Setting the environment (see page 26)
 - Setting the system PATH (see page 26)
 - Setting the classpath (see page 26)
 - Setting up the Java security manager for Java 2 (see page 31)
 - Installing a WebLogic license (see page 33)
3. Starting WebLogic Server
 - Important note regarding WebLogic RMI over IIOP (see page 40)
 - Starting WebLogic Server from the command line (see page 40)
 - Starting WebLogic Server from the WebLogic Console (see page 40)
 - Starting WebLogic Server using scripts (see page 46)
 - Starting WebLogic Server on Windows NT (see page 46)
 - Start Menu (see page 36)
 - NT Service (see page 36)
 - Windows Convenience Programs (see page 37)
 - Starting WebLogic Server from the command line (see page 40)
4. Next steps (see page 46)
 - Install JDBC drivers for use with WebLogic Server (see page 46)
 - Other Documentation (see page 47)

Install a JDK

WebLogic Server requires that you have a Java Development Kit (JDK) installed on your computer. A JDK provides a Java runtime environment (the Java Virtual Machine or JVM) and tools for compiling and debugging your Java applications. There are versions of the JDK for Windows NT and Solaris available free from JavaSoft and SunSoft. JDKs for other platforms are available from the platforms' manufacturers.

These Java environments are constantly under revision and are frequently updated with new releases and bug fixes. For more information on the latest JDKs supported for running WebLogic Server, check the [WebLogic Platform support](http://www.weblogic.com/docs51/platforms/index.html) page at <http://www.weblogic.com/docs51/platforms/index.html>.

If you installed WebLogic Server on Windows NT using InstallShield, a Java Runtime Environment (JRE) is automatically installed and configured to run WebLogic Server. A JRE provides only a runtime environment, and does not include development tools such as compilers and debuggers. Many Java applications developed for deployment with WebLogic Server require that a compiler be available while an application is running. For these applications, you will need to install a JDK.

Native vs. Green Threads

Most JDKs include an option to run the JVM using *native* or *green* threads. Always run WebLogic Server using *native* threads. Some JVMs, such as the JDK 1.2.2 for Solaris default to green threads. Usually passing the `-native` option on the `java` command line will force the JVM to use native threads. However, this implementation is not consistent across JDKs. Consult the documentation for your JDK to make sure that you correctly specify the use of native threads.

Hot Spot

Hot Spot is an enhancement to the standard JVM that uses a Just-in-time compiler and other features designed to improve performance. If you want to use a Hot Spot JVM, check the WebLogic Server platforms page to make sure that its use is supported for your platform.

Under most JDKs you can specify that the Hot Spot JVM not be used by adding the `-classic` option to the `java` command line. This can be useful if you require thread dumps while debugging an application. Because Hot Spot uses a JIT, thread dumps are not available.

Setting the environment

If you wish to run WebLogic Server from the command line or with scripts, follow the procedures in this section to set up your system PATH, Java system classpath, WebLogic classpath, `weblogic.properties` file, `weblogic.policy` file, and licenses.

Setting the system PATH

Set your system PATH to include the `java\bin` directory of your JDK and the `weblogic\bin` directory. If you are using the type-2 WebLogic jDriver for Oracle, you must also include the path to both the Oracle-supplied and the BEA-supplied client libraries for your driver. See [Installing WebLogic jDriver for Oracle](#) (on page 49) for more information. On Windows NT, you would use the following command to set your path in a command window:

```
$set path=c:\java\bin;c:\weblogic\bin;%path%
```

Where *java* is the directory containing your JDK and *weblogic* is the directory containing your WebLogic Server installation.

Setting the classpath

In the Java environment, the Java virtual machine uses the classpath to locate the classes it needs to run an application. Setting your classpath correctly is essential for running WebLogic Server or any Java application. WebLogic Server uses a process called *dynamic class loading* and therefore requires a combination of two settings for classpath:

- The command line **-classpath** option, which sets the **Java system classpath**. These are the classes WebLogic Server requires to start.
- A system property called **weblogic.class.path**, which sets the **WebLogic classpath** and specifies classes used while running WebLogic Server.

Although it is common practice for Java users to set the Java system classpath with the `CLASSPATH` environment variable, BEA recommends, when starting WebLogic Server, that you set your Java system classpath from the command line, using the `-classpath` option of the `java` command, which overrides any environment `CLASSPATH` setting. Using this procedure ensures that only the correct classes are loaded and that conflicting classes are not loaded.

Note: If you are using Microsoft SDK for Java (Jview) as your JDK, you must set the Java system classpath with the environment variable `CLASSPATH`. Using the `/cp` command line option to specify the Java system classpath will not allow WebLogic Server to start. For more information, see [Microsoft SDK for Java \(JView\)](#) on page 29.

These instructions apply only to running WebLogic Server. If you will be running any WebLogic examples, Java utilities, WebLogic clients, or other Java applications, you will need to set the classpath appropriately for those applications. For more information see [Setting classpath](#) at <http://www.weblogic.com/docs51/admindocs/classpath.html>. You can also find instructions for setting the classpath in the documentation for the examples and utilities.

Upgrading from a previous release

If you are upgrading from any earlier release of WebLogic Server, you should pay special attention to classpath issues because the requirements have changed. WebLogic Server now uses a new class loader to load classes. The class loader requires that you specify the classpath differently and also requires you to update any scripts or shortcuts you use to run WebLogic Server.

If you have Enterprise Java Beans compiled under an earlier release, you must perform several steps, including re-compiling, to update them for use with this release. For more information, see [Upgrading EJBs to WebLogic Server Version 5.1](#) at http://www.weblogic.com/docs51/classdocs/API_ejb/EJB_upgrade.html.

If you have any user-written or third party server-side classes, copy them to:

```
c:\weblogic\myserver\serverclasses
```

If you have any user-written or third party client-side classes, copy them to

```
c:\weblogic\myserver\clientclasses
```

Where `c:\weblogic` is the path to the directory where you installed WebLogic Server.

Setting your Java system classpath

The following must be included as arguments to the **-classpath** option on the **java** command line:

- `/java/lib/classes.zip`
Omit if you are using Java 2 (JDK 1.2.x)
- `/weblogic/classes/boot`
- `/weblogic/eval/cloudscape/lib/cloudscape.jar` (This entry is necessary only if you will be using the [Cloudscape DBMS](#).)
- `/weblogic/lib/poolorb.jar` (This entry is necessary only if you will be using WebLogic Enterprise Connectivity and JDK 1.2.x.)

Where `/weblogic` is the path to the directory where you installed WebLogic Server and `/java` is the path to the JDK directory.

Setting your WebLogic classpath

The following must be included as values for the **-Dweblogic.class.path** property. These classes will be loaded using the Weblogic classloader:

- `/weblogic/classes`
- `/weblogic/license`
- `/weblogic/lib/weblogicaux.jar`
- `/weblogic/myserver/serverclasses`
- The path to any other third-party or user-written classes.
Occasionally, you may encounter classes that cannot be placed in the WebLogic classpath. If you receive any unexplained “Class not found” errors, the problem may be that the class was written in such a way that it cannot be loaded with the class loader used by WebLogic Server. Such a class should be included in the Java system classpath as an additional argument to the **-classpath** option.

Example of setting classpath to run WebLogic Server under JDK 1.1.x (should be entered all on one line):

```
$ java -ms64m -mx64m -classpath
  c:/java/lib/classes.zip;
  c:/weblogic/classes/boot
  -Dweblogic.class.path=c:/weblogic/classes;
  c:/weblogic/license;
```

```
c:/weblogic/lib/weblogicaux.jar;  
c:/weblogic/myserver/serverclasses weblogic.Server
```

Where `c:/weblogic` is the path to the directory where you installed WebLogic Server and `c:/java` is the path to your JDK directory.

Verbose output of classloader

To see the location of classes loaded by the WebLogic classloader, you can set the following property in the command line or script used to start WebLogic Server:

```
-Dweblogic.classloader.verbose=true.
```

When this property is set, the location of loaded classes will be displayed as WebLogic Server starts up.

Microsoft SDK for Java (JView)

The command line syntax for `jview` is different from the `java` command. To run WebLogic Server under `jview`, make the following substitutions:

- `jview` for `java`
- `/d:` for the `-D` option (to set properties)

Note that `jview` does not support the class loader used by WebLogic Server. A WebLogic Server started under `jview` will not be able to use the class loader to load classes from the WebLogic classpath (the classes normally specified with the `weblogic.class.path` property). Since the class loader is not used, **you will not be able to use the Hot Deploy feature for deploying EJBs or servlets in a running WebLogic Server without having to restart the server.**

Specify the following special property when starting the server under `jview`. This property prevents WebLogic Server from using the `weblogic.class.path` property to load classes:

```
/d:weblogic.system.disableWeblogicClassPath=true
```

To run WebLogic Server under `jview`, do not specify the `weblogic.class.path` property on the command line. Instead, specify those classes that would otherwise go in the WebLogic classpath in the Java system classpath, with the environment variable `CLASSPATH`, along with the classes which normally belong in the Java system classpath.

```
jview /d:weblogic.system.disableWeblogicClassPath=true  
weblogic.Server
```

Where `c:\weblogic` is the path to the directory where you installed WebLogic Server.

Using jview with WebLogic COM

If you are using WebLogic COM, also set your trusted classpath. See [Using WebLogic COM](http://www.weblogic.com/docs51/classdocs/API_com.html) at http://www.weblogic.com/docs51/classdocs/API_com.html for more information.

The WebLogic Frequently Asked Questions also has a section on [jview](http://www.weblogic.com/docs51/techsupport/faq/3rdparty.html#jview) at <http://www.weblogic.com/docs51/techsupport/faq/3rdparty.html#jview>.

Using jview with RMI

If you will be using RMI with jview, you must add the following zip file to your Java system classpath (by adding it to the `CLASSPATH` environment variable) when starting WebLogic Server:

```
weblogic/lib/rmiForMs.zip
```

Where `weblogic` is the directory where you installed WebLogic Server

Starting WebLogic Server Statically

You can start WebLogic Server statically, without using the `weblogic.class.path` to specify its classes. This can be useful when using some Integrated Development Environments (IDE) or when running a debugger. *However, when you start WebLogic Server statically, you can not deploy EJBs or servlets in a running WebLogic Server (re-starting WebLogic Server is required).*

To start WebLogic Server statically:

1. Do not use the `weblogic.class.path` property on the command line. Instead, specify all of the classes required to start WebLogic Server and run your applications using the environment variable `CLASSPATH`. This includes the classes described in this document as belonging in the Java system classpath *and* those belonging in the WebLogic classpath.

2. Add the following property on the Java command line you use to start WebLogic Server:

```
-Dweblogic.system.disableWeblogicClassPath=true
```

This property prevents WebLogic Server from using the `weblogic.class.path` property to load classes.

Cloudscape DBMS

WebLogic Server comes with a trial version of an all-Java database management system (DBMS) called **Cloudscape**. The WebLogic Tour and some of the example code shipped with WebLogic Server use this DBMS. You can also use it for testing if you do not have another DBMS available. If you will be using Cloudscape, you must include it in your Java system classpath. Normally third-party classes such as these should be included in the WebLogic classpath (with the `weblogic.class.path` property). However, due to some differences in the Cloudscape product, this jar file should be included in the Java system classpath (using the `-classpath` option). For additional information, see [Using the Cloudscape database with WebLogic](http://www.weblogic.com/docs51/techsupport/cloudscape.html) at <http://www.weblogic.com/docs51/techsupport/cloudscape.html>.

Setting up the Java security manager for Java 2

When you run WebLogic Server under Java 2 (JDK 1.2.x), the server uses a **Java Security Manager** to control access to system resources. Java Security Manager requires a **security policy** file to set up the permissions. The WebLogic distribution contains a security policy file (called `weblogic.policy`) that contains a set of default permissions that allows you to start WebLogic Server without creating your own security policy.

Modifying the `weblogic.policy` file for general use

Windows NT InstallShield users may skip the remainder of this section. InstallShield modifies the `weblogic.policy` file automatically.

To modify the `weblogic.policy` file included with your distribution:

1. Edit the following two lines in the `weblogic.policy` file, changing the items in **bold** to match the location of the directory where you installed WebLogic Server:

```
grant codeBase "file:/c:/weblogic/" {  
  
permission java.io.FilePermission "c:${/}weblogic${/}-", ...
```

2. Set these two properties on the Java command line when you start WebLogic Server:

- `java.security.manager` tells the JVM to use a security policy. You do not need to specify any arguments to this property.
- `java.security.policy` tells the JVM the location of the security policy file to use. The argument is the fully qualified file name of the security policy file, in this case, `weblogic.policy`.

For example,

```
$ java ... -Djava.security.manager  
          -Djava.security.policy==c:/weblogic/weblogic.policy
```

Be sure to use “==” instead of “=” when specifying `java.security.policy`, so that only the `weblogic.policy` file is used by Java security manager. The == causes the `weblogic.policy` file to override any default security policy. A single equal sign causes the `weblogic.policy` file to be appended to an existing security policy. For more information on setting up a security policy, see the article [Default Policy Implementation and Policy File Syntax](http://java.sun.com/products/jdk/1.2/docs/guide/security/PolicyFiles.html) on the JavaSoft website at

<http://java.sun.com/products/jdk/1.2/docs/guide/security/PolicyFiles.html>.

Modifying the `weblogic.policy` file for third party or user-written classes

The best location for your server-side user code is the `weblogic/myserver/serverclasses` directory. If you have third party or user-written classes that are not in that directory, also:

1. Copy the entire block of code in the `weblogic.policy` file from “grant codebase ...” to the closing bracket and semicolon.
2. Paste the selection back into the `weblogic.policy` file below the section you just copied.

3. Edit the `grant codeBase` and the `permission java.io.FilePermission` statements so that the directories point to the location of your third party or user-written code.

This procedure creates a security policy for your code that contains exactly the same permissions as those for the WebLogic Server. You should examine these permissions closely to make sure that this is the security policy you want to use for those directories. For more information on setting up a security policy, see the article [Default Policy Implementation and Policy File Syntax](#) on the JavaSoft website at

<http://java.sun.com/products/jdk/1.2/docs/guide/security/PolicyFiles.html>.

Caution: Using JavaSoft JDK version 1.2.1 on UNIX systems applies security policy improperly if your WebLogic software is not installed in the root directory of the file system or disk drive. Policy is only applied correctly if the path in a `grant codeBase` URL has just one component. For example, if you install WebLogic Server in `c:\test\weblogic`, (or even `/home/weblogic` on Solaris), you will see `AccessControlExceptions` even though you use the correct URL in your policy file.

To work around this limitation, you can either install WebLogic in the root directory (recommended) or modify the URL so that it contains only the first component of the path to your WebLogic installation, for example:

```
grant codeBase "file:/c:/test/-" {
```

This problem has been acknowledged by Sun Microsystems as bug # 4261298.

Installing a WebLogic license

Your WebLogic distribution requires a valid license to run. This section tells you how to install and update WebLogic licenses.

Evaluation licenses

An evaluation copy of WebLogic Server is enabled for 30 days so you can start using WebLogic Server immediately. To use WebLogic Server beyond the 30-day evaluation period or to use [clustering](#) (see <http://www.weblogic.com/docs51/cluster/index.html> for more information) features,

you will need to contact your salesperson about further evaluation or purchasing a license for each IP address on which you intend to use WebLogic Server. All WebLogic Server evaluation products are licensed for use on a single server with access allowed from up to 3 unique client IP addresses.

If you downloaded WebLogic Server from the BEA website, your evaluation license is included with the distribution. InstallShield users will also receive a password by email. The InstallShield program will prompt you for this password during the installation process.

If you obtained WebLogic Server from an evaluation CD, follow the instructions on the CD for obtaining an evaluation license.

Other licenses

When you purchase a license for WebLogic Server you will receive a set of license keys by email. To use these keys, follow the instructions below, under Updating a license.

Updating a license

You need to add new keys to your existing license file if you have purchased more software, if you have applied for and received an extension to your 30-day evaluation, or if you get a new distribution that includes new products. You will receive a set of permanent, non-expiring license keys by email as an attachment after purchase. If you are evaluating WebLogic Server's clustering feature, you may receive a set of evaluation keys for clustering.

To add new license keys to your existing license file:

1. Open the message containing the keys you received by email when you purchased a license from BEA Systems, Inc.
2. Open the `WebLogicLicense.xml` file in a text editor. (Make sure you don't edit this file in Microsoft Word, or any other word-processing program that will save the file as a binary.)
3. Copy the keys from the email and paste them at the top of the XML file, after the `<WEBLOGIC-LICENSES>` tag.

For example, a key for clustering looks something like this:

```
<WEBLOGIC-LICENSES>
```



```
<LICENSE PRODUCT="WebLogic/ClusterII"
  IP="000.000.900.900"
  UNITS=" 5"
  EXPIRATION="31-Mar-2001"
  KEY="w20f8s08480v0adpup3485paprtnp8ac"
/>

... </WEBLOGIC-LICENSES>
```

4. Save the `WebLogicLicense.xml` file into your WebLogic license directory. If you have installed with defaults, that will be `/weblogic/license`.
5. Save your license key information and a copy of your `WebLogicLicense.xml` file in a safe place outside the WebLogic distribution. Although no one else will be able to use your license keys, you should probably save this information in a place protected from either malicious or innocent tampering by others. When you upgrade WebLogic Server, keep your original `WebLogicLicense.xml` and add new keys to it for any new services you purchase.

Upgrading licenses from a previous release

Prior to release 4.0 of WebLogic Server, licenses were distributed in a compiled Java `.class` format. WebLogic licenses are now distributed in an XML format. If you are upgrading from an earlier release of WebLogic Server, please copy your license files, called either `WebLogicLicense.XML` or `WebLogicLicense.class` to your `weblogic/license` directory. (Where *weblogic* is the directory containing your WebLogic Server installation.)

If you purchase more than one product from WebLogic, you will have more than one entry in your license file. If you have purchased licenses for versions previous to 4.0, you may also have a `.class` format license file. All of your license files can be used together. When the WebLogic Server starts up, it looks for multiple license files in both formats in the following order:

1. `.class` files in the `weblogic/license` directory
2. `.XML` files in the directory specified with the `weblogic.system.home` property
3. `.XML` files in the WebLogic Server classpath.

If WebLogic Server encounters an expired license, it will *not* continue to look for additional licenses. For this reason you should remove expired license keys from your license files. (If you are using a `.class` license file, you must recompile the `WebLogicLicense.java` file after removing the expired license.)

Always restart WebLogic Server after making any changes to your license files.

For users who have `.class` license files, BEA recommends that you use the [convertLicense utility](http://www.weblogic.com/docs51/techstart/utls.html#licenseConverter) (see <http://www.weblogic.com/docs51/techstart/utls.html#licenseConverter>) to convert your license file to an XML format license. You can then combine your XML licenses into a single file by cutting and pasting between the two files. For information on editing an XML style license, see [Installing a WebLogic License](http://www.weblogic.com/docs51/techstart/license.html) at <http://www.weblogic.com/docs51/techstart/license.html>.

If you need to add license keys to an older class-style license, please see [Installing a WebLogic License](http://www.weblogic.com/docs51/techstart/license.html) at <http://www.weblogic.com/docs51/techstart/license.html>.

Starting WebLogic Server on Windows NT

Start Menu

If you installed WebLogic Server on Windows with the InstallShield kit, you can use the WebLogic Server shortcut on the Windows start menu to start the WebLogic Server. Click on:

Start—>Programs—>WebLogic 5.1—>WebLogic Server.

You can use the `wlconfig` utility to select the various defaults used when starting WebLogic Server from the Start menu. See [Windows Convenience Programs](#) on page 37 for more information about this utility.

NT Service

You can also run WebLogic Server as a Windows NT service. When installed as an NT service, WebLogic Server will start automatically when you boot the Windows NT computer. A WebLogic Server started in this way will use the same start up parameters (stored in the Windows Registry) that are used when starting a WebLogic Server using

the Windows start menu or the Windows convenience program `wlserver.exe`. These parameters may be changed by using the `wlconfig.exe` program, described under Windows Convenience Programs on page 37.

For additional information, see Using WebLogic Server as an NT 4.0 service at <http://www.weblogic.com/docs51/admindocs/ntservice.html>.

You must have administrator-level privileges to either install or uninstall an application as an NT service.

To install WebLogic Server as a Windows NT service:

1. Switch to the `weblogic\bin` directory.

2. Run `install.exe`. For example,

```
c:\weblogic\bin> install.exe
```

3. If you wish to run multiple instances of WebLogic Server as NT services, name the instances by adding an additional parameter specifying the name for each server instance. For example:

```
c:\weblogic\bin> install -name thisWebLogicServer
```

Multiple instances of WebLogic Server can be useful when testing WebLogic Clusters on a single computer. There is additional information available on Installing and removing multiple WebLogic NT services at <http://www.weblogic.com/docs51/admindocs/ntservice.html#multiple>.

Windows Convenience Programs

The following programs (Windows NT only) may be run from the `weblogic/bin` directory:

Note: The names of these programs changed as of Version 5.0 of WebLogic Server:

```
t3config is now wlconfig
t3server is now wlserver
t3console is now wlconsole
```

dbping.exe

Tests your connection to a database.

Arguments:

DBMS

Can be one of the following:

ORACLE
MSSQLSERVER4
INFORMIX4

user

Valid username for database login. Use the same values and format that you use with isql for SQL Server, sqlplus for Oracle, or DBACCESS for Informix.

password

Valid password for the user. Use the same values and format that you use with isql, sqlplus, or DBACCESS.

DB

Name of the database. The format varies depending on the database and version. Use the same values and format that you use with isql, sqlplus, or DBACCESS. Type 4 drivers, such as MSSQLServer4 and Informix4, need additional information to locate the server since they cannot access the environment.

For example:

```
$ dbping ORACLE scott tiger demo
```

install.exe

Installs WebLogic Server to run as a Windows NT service.

remove.exe

Removes WebLogic Server as a Windows NT service.

wlconfig.exe

Makes changes to your installation configuration. Note that **wlconfig.exe** changes settings in the Windows NT registry and therefore these settings will remain the defaults until you run **wlconfig.exe** again. These settings are used when running the **wlserver.exe** convenience program (see below), when running WebLogic Server as an NT service, and when running WebLogic Server from the start menu.

The following options are available for **wlconfig.exe**. If no options are specified, **wlconfig.exe** prints a list of the current settings.

`-help`

Prints a list of optional arguments for the `wlconfig.exe` command.

`-msSizeInMb`

Sets the initial Java heap size (in megabytes).

See [WebLogic Server Performance Tuning Guide](#) at

<http://www.weblogic.com/docs51/admindocs/tuning.html#jvm>
execution for information on setting heap size.

`-mxSizeInMb`

Sets the maximum Java heap size (in megabytes).

See [WebLogic Server Performance Tuning Guide](#) at

<http://www.weblogic.com/docs51/admindocs/tuning.html#jvm>
execution for information on setting heap size.

`-JAVA_HOME path`

Changes the JDK/JRE runtime directory on which the WebLogic Server installation depends. The default points to the directory where you installed the WebLogic distribution (for example, `c:\weblogic`), which is packaged with its own JRE.

`-classpath path`

Changes the system classpath on which the WebLogic Server installation depends. The default is unset; that is there are no directories in the classpath.

`-Dproperty=value`

Use the `-D` command to change other properties. These properties are stored in the Windows Registry and are used as command line arguments when starting the server using `wlserver.exe`, the Start menu, and when starting as an NT service.

wlconsole.exe

Runs the WebLogic Console, an administrative tool.

wlserver.exe

Runs WebLogic Server with the defaults defined by running `wlconfig.exe` (see above).

version.exe

Displays the current version of your WebLogic Server.

Starting WebLogic Server from the command line

The WebLogic Server is a Java class file, and like any Java application, you can start it with the `java` command. These instructions describe the options you need to include when starting WebLogic Server from the command line. The startup options described here will start WebLogic Server in a minimal configuration. To use any of the APIs or services of WebLogic Server, you should consult the [Developers Guides](#), [API Reference](#), and [Deployment Guides](#) included with the WebLogic Server documentation (available at <http://www.weblogic.com/docs50>).

You will notice that the command lines required to start WebLogic Server can be quite lengthy and tedious to type. To make sure that your start-up commands are accurate, BEA recommends that you incorporate these command lines into scripts that you can then use to start WebLogic Server. For more information, see [Starting WebLogic Server using scripts](#) on page 46.

Important note regarding WebLogic RMI over IIOP

If you are using **WebLogic RMI over IIOP**, see [Using WebLogic RMI over IIOP](#) at http://www.weblogic.com/docs51/classdocs/API_rmi_iiop.html#StartServer.

Requirements for Starting WebLogic Server

The following are required when starting WebLogic Server:

- Start WebLogic Server with the Java `-ms64m` and `-mx64m` options. These options allocate a minimum and maximum of 64 megabytes of Java heap memory to the WebLogic Server. These values assigned to these parameters can dramatically effect the performance of your WebLogic Server and are provided here only as general defaults. In a production environment you should carefully consider the correct memory heap size to use for your applications and environment. For more information on setting heap size, See [WebLogic Server Performance](#).

Tuning Guide at http://www.weblogic.com/docs51/admindocs/tuning.html#jvm_execution.

- Set your **-classpath** option and **weblogic.class.path** as shown in Setting your WebLogic classpath on page 28. These are the minimum requirements for classpath:

- `-classpath c:/java/lib/classes.zip;c:/weblogic/classes/boot`

If you are using Java 2, omit `c:/java/lib/classes.zip` from the `-classpath` option.

- `-Dweblogic.class.path=c:/weblogic/classes;
c:/weblogic/license;c:/weblogic/lib/weblogicaux.jar;
c:/weblogic/myserver/serverclasses`

Where `c:/weblogic` is the directory where you installed WebLogic Server and `c:/java` is the path to your JDK.

- If you are using Java 2, set the **java.security.manager** property and the **java.security.policy** property to point to the location of the `weblogic.policy` file. (See Setting up the Java security manager for Java 2 on page 31 for information on setting up your `weblogic.policy` file).

- If you are not starting WebLogic Server from the installation directory, add

`-Dweblogic.system.home=c:/weblogic`

Where `c:/weblogic` is the directory containing your `weblogic.properties` file. (This is usually the same as the directory where you installed WebLogic Server.)

- If you are using **third-party container managed persistence**, add the following property to the command line:

`-Dweblogic.home=weblogic`

Where `weblogic` is the path to the directory where you installed WebLogic Server. Note that this property is different than the `weblogic.system.home` property (see <http://www.weblogic.com/docs51/admindocs/properties.html#P2> for more information).

Command line examples

Here are some sample command lines you can use to start WebLogic Server. These examples assume that you installed WebLogic Server in the `c:/weblogic` directory and that your JDK 1.1 is located in the `c:/java` directory. Modify these commands, substituting the correct directories for your installation.

The samples also assume that you are starting WebLogic Server from the installed directory. If you are starting from a different directory, add the following property to the command line, substituting the directory containing your WebLogic Server installation:

```
-Dweblogic.system.home=c:/weblogic
```

Where `c:/weblogic` is the directory containing your `weblogic.properties` file. (This is usually the same as the directory where you installed WebLogic Server.)

Although these examples are broken into multiple lines for readability, the commands should be entered as one line.

JDK 1.1.x example

```
$ java -ms64m -mx64m -classpath  
c:/java/lib/classes.zip;  
c:/weblogic/classes/boot  
-Dweblogic.class.path=c:/weblogic/classes;  
c:/weblogic/license;c:/weblogic/lib/weblogicaux.jar;  
c:/weblogic/myserver/serverclasses weblogic.Server
```

JDK 1.2 (Java 2) example

```
$ java -ms64m -mx64m -classpath c:/weblogic/classes/boot  
-Dweblogic.class.path=c:/weblogic/classes;  
c:/weblogic/license;c:/weblogic/lib/weblogicaux.jar;  
c:/weblogic/myserver/serverclasses  
-Djava.security.manager  
-Djava.security.policy==c:/weblogic/weblogic.policy  
weblogic.Server
```


Jview example

```
$ jview /d:weblogic.system.disableWeblogicClassPath=true  
weblogic.Server
```

There is important information you should be aware of when using Jview. Please read the section [Microsoft SDK for Java \(JView\)](#) on page 29.

Additional options:

- If you are not starting WebLogic Server from its installed directory, add:
`-Dweblogic.system.home=c:/weblogic` (or the directory containing your `weblogic.properties` file)
- If you will be using the Cloudscape DBMS, add:
`c:\weblogic\eval\cloudscape\lib\cloudscape.jar` to the `-classpath` option.
- Add the location of any third-party or user-written classes to the `weblogic.class.path` property. (When running under jview, place these in the Java system classpath, with the `/cp` option.)

Starting WebLogic Enterprise Connectivity

To start WebLogic Enterprise Connectivity with JDK 1.2 (Java 2), add the following to the Java system classpath:

```
c:/weblogic/lib/poolorb.jar
```

For more information, see the Developers Guide [Using WebLogic Enterprise Connectivity](#) at http://www.weblogic.com/docs51/classdocs/API_wlec.html.

Starting WebLogic Server from the WebLogic Console

The WebLogic Console is a pure-Java GUI management console where you can monitor WebLogic Server performance and other aspects of the WebLogic Server's environment. You can also use the console to start WebLogic Server. For more information, see [Running the WebLogic Console](http://www.weblogic.com/docs51/admindocs/console.html) at <http://www.weblogic.com/docs51/admindocs/console.html>.

To start WebLogic Server from the console:

1. Start the console from the Windows NT start menu by selecting **Start—>Programs—>WebLogic—>WebLogic Console**, or use the following command:

```
$ java -mx32m -classpath c:/java/lib/classes.zip;  
c:/weblogic/classes;  
c:/weblogic/lib/weblogicaux.jar weblogic.Console
```

Where *c:/java* is the path to your JDK (this may be omitted when running under Java 2) and *c:/weblogic* is the path to your WebLogic Server installation.

You can also start the WebLogic Console with the supplied scripts, **startConsole.sh** (UNIX) and **startConsole.cmd** (Windows NT), and **startConsoleJview.cmd** (Windows NT running under Microsoft SDK for Java). These scripts are located in the root directory of your WebLogic distribution.

You will need to modify these scripts for your environment. See [Starting WebLogic Server using scripts](#) on page 46.

2. From the console menu bar, click on:

File —>Start a new WebLogic Server or Cluster.

A dialog box will appear. Fill in the following information:

Name of the server

The name of the WebLogic Server you are starting. This parameter is used when identifying WebLogic Servers within a WebLogic

Cluster. The default name for starting a single WebLogic Server is **myserver**.

WebLogic home

The directory where you installed WebLogic Server.

Enable cluster

Check this box if you are starting a WebLogic Server as part of a WebLogic Cluster.

Cluster name

The name of the WebLogic Cluster that this WebLogic Server will join.

Advanced

The **Advanced** button opens a dialog box where you can set the Listen port, SSL listen port, Java heap size, Multicast address, and bind address.

3. A dialog box will pop up saying that WebLogic Server has started successfully. Click OK.
4. Another dialog box labeled “Attach to WebLogic running WebLogic server” will appear. Fill in the following information:

User

For privileged, administration-level access the user is always **system**.

Password

This is the password you entered when installing WebLogic Server. It is stored in the `weblogic.properties` file under the property `weblogic.password.system`.

DNS host name

The host name of the machine running WebLogic Server. The default is **localhost**.

Starting WebLogic Server using scripts

Sample scripts are provided with the WebLogic distribution that you can use to start WebLogic Server. You will need to modify these scripts to fit your environment and applications: The scripts are called `startWebLogic.sh` (UNIX) and `startWeblogic.cmd` (Windows NT). These scripts are located in the root directory of your WebLogic distribution.

To use the supplied scripts:

- Pay close attention to classpath settings and directory names.
- Change the value of the variable `JAVA_HOME` to the location of your JDK.
- UNIX users must change the permissions of the sample UNIX script to make the file executable. For example:

```
chmod +x startWebLogic.sh
```

Next steps

Set up your development environment

Scripts called `setEnv.cmd` (Windows NT) or `setEnv.sh` (UNIX) are included in the root directory of your WebLogic Server installation. These scripts will set up the appropriate environment for development and running the code examples included with WebLogic Server. You will need to modify these scripts somewhat for your environment. For more information, see [Setting your development environment](http://www.weblogic.com/docs51/techstart/environment.html) at <http://www.weblogic.com/docs51/techstart/environment.html>.

Install JDBC drivers for use with WebLogic Server

If you will be using a JDBC driver for database access, see the following links:

Oracle

Installing WebLogic jDriver for Oracle on page 49. Users upgrading from an earlier release of WebLogic Server should pay special attention to their Oracle configuration. There are now several versions of this driver available and these additions require that you set your PATH (Windows NT) or shared library path (Unix) differently.

Using the Oracle thin driver with WebLogic Server at <http://www.weblogic.com/docs51/classdocs/oraclethin.html> contains information on using Oracle's thin driver (available from Oracle).

Informix

Installing WebLogic jDriver for Informix at http://www.weblogic.com/docs51/techstart/install_jinf4.html.

Microsoft SQL Server

Installing WebLogic jDriver for Microsoft SQL Server at http://www.weblogic.com/docs51/techstart/install_jmsq4.lhtml.

Sybase

The jConnect JDBC driver from Sybase is now bundled with WebLogic Server. For information on using this driver with WebLogic Server, see Using the Sybase jConnect driver at <http://www.weblogic.com/docs51/classdocs/jConnect.html>.

Other Documentation

- Using WebLogic Clusters at <http://www.weblogic.com/docs51/cluster/index.html>.
- WebLogic Developer Center (<http://www.weblogic.com/docs51/resources.html>)
- WebLogic Administrators Guides (<http://www.weblogic.com/docs51/admindocs/index.html>)

- WebLogic Developers Guides
(<http://www.weblogic.com/docs51/classdocs/index.html>)
- WebLogic API Reference
(<http://www.weblogic.com/docs51/classdocs/packages.html>)
- WebLogic Server comes with many code examples to help you get started. See the Guide to the WebLogic code examples at <http://www.weblogic.com/docs51/examples/index.html>.

5 Installing WebLogic jDriver for Oracle

Overview

WebLogic jDriver for Oracle is a new name for the product formerly known as jdbcKona/Oracle.

WebLogic jDriver for Oracle, a Type-2 JDBC driver for the Oracle DBMS is included with WebLogic Server. For this driver to function, you must have a complete Oracle client installed on the machine that will be the client to the Oracle DBMS. This Oracle client installation contains vendor-supplied client libraries and associated files that WebLogic jDriver for Oracle requires to operate.

The WebLogic Server distribution includes a choice of several BEA-supplied native libraries for WebLogic jDriver for Oracle. The library that you choose depends on which Oracle client version you have installed on your client machine and which version of the Oracle API you will use to access your Oracle server. Installing this driver requires that both the BEA-supplied native library and the Oracle-supplied client libraries be available by including them in your the client's `PATH` (Windows NT) or `shared library path` (UNIX), as described below.

For information on using WebLogic jDriver for Oracle see [Using WebLogic jDriver for Oracle](http://www.weblogic.com/docs51/classdocs/API_joci.html) (at http://www.weblogic.com/docs51/classdocs/API_joci.html).

Important issues for release 5.1

Please read the Release notes for details on the issues highlighted below.

Oracle Release Notes are also available on line at
http://www.weblogic.com/docs51/classdocs/release_notes_joci.html.

Platform Support:

Check the Release Notes or the WebLogic Platform Support page (at <http://www.weblogic.com/docs51/platforms/index.html>) in the online documentation for details about which platforms, operating systems, DBMS versions, and Java versions are supported for WebLogic jDriver for Oracle.

Important issues include:

- WebLogic jDriver for Oracle is not supported when using the Oracle 8 API and connecting to an version 7 Oracle DBMS.
- Using `CallableStatement.getResultSet()`
- Mixing different versions of the Oracle Client and Oracle Server
- Codeset conversion with CLOBs when using a different operating system on the client and server.

Other JDBC drivers

BEA also has Type-4 JDBC drivers available for the Informix and Microsoft SQL Server DBMSs. These are pure-Java drivers and do not require vendor-supplied client libraries.

For more information on these drivers, please see: WebLogic JDBC Options, at <http://www.weblogic.com/docs51/classdocs/jdbcdrivers.html>.

Installation steps

1. **Select the appropriate native library for your environment.**

WebLogic jDriver for Oracle is shipped with `dll`, `sl`, or `so` files for various combinations of Oracle client versions, APIs, and platforms. You must place the appropriate file in your system's path or shared library path for this driver to function. *Unlike previous releases of WebLogic jDriver for Oracle, which had only one such file that was always present in the default path, you now must specify this path setting explicitly.* For details and instructions on setting these options, see [Setting your path and client libraries](#) on page 54.

2. Add the client libraries from your Oracle installation to your system PATH (Windows NT) or load library path (UNIX—the name of this variable differs among UNIX systems). Check your Oracle documentation for the location of these libraries. On Windows NT the client libraries are usually located in `c:\ORANT\bin`.

3. **Unpack the distribution.**

If you have purchased WebLogic Server, WebLogic jDriver for Oracle is included with your distribution. No further steps are required to unpack the distribution. The remainder of these installation steps are required only for users of the stand-alone version of WebLogic jDriver for Oracle.

If you haven't already, unpack the archive you downloaded *in the root directory* of your computer. You must use a program such as [WinZip](http://www.winzip.com) at <http://www.winzip.com> that preserves the directory structure of the archive. The files unpack into the `weblogic` directory.

The instructions in this document assume that you are installing WebLogic jDriver for Oracle on a Windows computer and that you unpacked the archive in the root directory of your C drive. If you're installing on a different operating system or in a different location, check your system documentation for help in adjusting the commands shown in these instructions for your own environment.

4. **Add the WebLogic jDriver for Oracle classes directory to your CLASSPATH variable.**

Note: If you are using the WebLogic jDriver for Oracle that is bundled with WebLogic Server, the classpath setting are the same as those required for

WebLogic Server. The standard start up scripts supplied with the WebLogic Server distribution will do this for you.

To change your CLASSPATH temporarily at a Windows Command Prompt, use this command:

```
$ set CLASSPATH=%CLASSPATH%;c:\weblogic\informix4\classes
```

To change your CLASSPATH permanently:

- a. Double-click the System icon in the Control Panel.
- b. Click the Environment tab.
- c. In the lower panel, select the CLASSPATH variable. If no CLASSPATH variable is defined, add it.
- d. Add c:\weblogic\informix4\classes to the value of the CLASSPATH variable. Use a semicolon (;) to separate the new path from the previous value of the variable, if any.

Any program you launch after you update the CLASSPATH variable in the System control panel will have access to the new value.

For more help setting your CLASSPATH, read [Setting classpath](http://www.weblogic.com/docs51/admindocs/classpath.html) at <http://www.weblogic.com/docs51/admindocs/classpath.html>. If you're using WebLogic jDriver for Oracle from within an IDE like Symantec Cafe or JBuilder, the procedure for adding classes to the CLASSPATH may be different.

5. **Install WebLogic jDriver for Oracle for use with browser applets.**

If you're planning to use WebLogic jDriver for Oracle to access databases via applets in a web browser, you must install the distribution on the computer executing the web server. Furthermore, the web server and the Informix database server must be running on the same computer. This is a security restriction enforced by most web browsers, including Netscape Navigator and Internet Explorer. For information about browser security and applets, read [Troubleshooting Applet Security problems](http://www.weblogic.com/docs51/techsupport/appletsecurity.html) at <http://www.weblogic.com/docs51/techsupport/appletsecurity.html>.

The web server must also be able to find the WebLogic jDriver for Oracle class files. To ensure that the web server can find the files, install the WebLogic jDriver for Oracle distribution on your web server computer and then set the CLASSPATH on that computer as described in step 4. Be sure to restart the web server after you change the CLASSPATH so that the web server has access to the revised CLASSPATH value.

6. Install a license file.

With the release of version 5.1 of WebLogic jDriver for Oracle, a new type of license is shipped with the distribution. This license uses an XML file to store the license keys.

There are several issues to consider regarding your license for WebLogic jDriver for Oracle:

- If you downloaded an evaluation version of the driver, your distribution includes an evaluation license. Your driver is ready for evaluation use.
- If you have previously purchased a license for WebLogic jDriver for Oracle, *you must obtain an updated license file from your BEA contact person* and place that license in your WebLogic jDriver for Oracle installation.

Copy your license key into your `WebLogicLicense.xml` file. This file is located at `weblogic/informix4/license/WebLogicLicense.xml` (where *weblogic* is the directory containing your WebLogic jDriver for Oracle installation). For instructions on editing a WebLogic XML license file, see [Editing an entry to the XML license file](#), below.

To purchase a permanent license for WebLogic jDriver for Oracle, contact sales@weblogic.com.

Editing an entry to the XML license file

1. Open the message containing the key(s) you received by email when you purchased a license from BEA Systems, Inc.
2. Open the `WebLogicLicense.xml` file in a text editor. This file is located in one of the following directories:
 - standalone driver: `weblogic/informix4/license`
 - using the driver with WebLogic Server: `weblogic/license`

(Where *weblogic* is the directory containing your WebLogic jDriver for Oracle or WebLogic Server installation.)

Do not edit this file in Microsoft Word or any other word-processing program that will save the file as a binary.

3. Copy the keys from the email and paste them **at the top** of the XML file.

For example, if you have received a key, it'll look something like this:

```
<LICENSE PRODUCT="jdbcKona/Informix4"
      IP="000.000.900.900"
      UNITS="5"
      EXPIRATION="31-Mar-1999"
      KEY="w20f8s08480v0adpup43245paprtnp8ac"
/>
```

4. Save the `WebLogicLicense.xml` file into the same directory.
5. Save your license key information and a copy of your `WebLogicLicense.xml` file in a safe place outside the WebLogic distribution, preferably someplace you will remember. Although no one else will be able to use your license keys, you should probably save this information in a place protected from either malicious or innocent tampering by others. Please note that when you upgrade your WebLogic classes, you will want to keep your original `WebLogicLicense.xml` and add new keys to it for any new services you purchase.

Setting your path and client libraries

A native `dll`, `so`, or `sl` file containing your driver must be made available to your WebLogic jDriver for Oracle client by including its directory in your system `PATH` (Windows NT), `LD_LIBRARY_PATH` (most Unix systems), or `SHLIB_PATH` (HP-UX). The directory containing the correct file varies depending on several factors discussed below.

The vendor-supplied libraries from Oracle must also be included in your `PATH`, `load library path`, or `shared library path`. The location of the directory containing your Oracle client libraries will vary depending on your installation. On Windows NT, these libraries are normally placed into your machine's path by the Oracle installer.

Version 5.1 of WebLogic jDriver for Oracle now uses the `dll`, `so`, or `sl` files built with the Oracle 8 API as the native interface for accessing an Oracle DBMS. This API allows for improved connectivity to a version 7 or version 8 Oracle Server and also provides access to new features available only in Oracle 8 (some of these features require JDBC 2.0, which requires a Java 2 JVM).

An older version of these `dll`, `so`, or `sl` files that uses the Oracle 7 API, is also included with this release to assure backwards compatibility.

The tables below, under Directory to put in your system PATH, list the Oracle client version, the Oracle API version and the directory you must put in your system PATH to access that version of the driver.

JDBC 2.0

JDBC 2.0 features are only available when using a driver that uses the Oracle 8 API. You must also run your WebLogic jDriver for Oracle client under a Java 2 compatible JDK. Using JDBC 2.0 also requires using a different driver class and URL in your Java code. For more information, see JDBC 2.0 for Oracle in the document Using WebLogic jDriver for Oracle at http://www.weblogic.com/docs51/classdocs/API_joci.html#jdbc20.

Platform considerations

For information on supported platforms and JVM versions for WebLogic jDriver for Oracle, see the WebLogic platform support page at <http://www.weblogic.com/docs51/platforms/index.html#jdbc>.

The minimum supported client library is Oracle version 7.3.4.

Directory to put in your system PATH

Windows NT

Add `weblogic\bin` and the appropriate directory from the table below to your PATH, for example:

```
$ set PATH=%PATH%;c:\weblogic\bin\oci805_8
```

Where `c:\weblogic` is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation.

Directory to put in your `PATH`

Oracle Client version	Oracle API version	Directory	Notes
7.3.4	7	oci734_7	Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.
8.0.5	7	oci805_7	Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.
8.0.5	8	oci805_8	Allows access to Oracle 8 and JDBC 2.0 features.
8.1.5	8	oci815_8	Allows access to Oracle 8 and JDBC 2.0 features.

Solaris

Add `weblogic/lib/solaris` and the appropriate directory from the table below to your `ld_library_path`. (Where `weblogic` is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation)

Directory to put in your `ld_library_path`

Oracle Client version	Oracle API version	Directory	Notes
7.3.4	7	oci734_7	Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.
8.0.5	7	oci805_7	Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.
8.0.5	8	oci805_8	Allows access to Oracle 8 and JDBC 2.0 features.
8.0.6	8	oci806_8	Allows access to Oracle 8 and JDBC 2.0 features.

8.1.5	8	oci815_8	Allows access to Oracle 8 and JDBC 2.0 features.
-------	---	----------	--

You must also include the path to your vendor-supplied client libraries from Oracle in your `ld_library_path`. The location of these libraries will depend on your Oracle client installation.

IBM AIX

WebLogic jDriver for Oracle for IBM AIX is available only for the Oracle 7 API. For a list of supported client versions, see [Platform support for JDBC drivers](http://www.weblogic.com/docs51/platforms/index.html#jdbc) at <http://www.weblogic.com/docs51/platforms/index.html#jdbc>.

Add the directory `weblogic/lib/aix` to your **LIBPATH**.

Where `weblogic` is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation.

HP-UX 11

Add `weblogic/lib/hpux11` and the appropriate directory from the table below to your **SHLIB_PATH**. (Where `weblogic` is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation)

Oracle Client version	Oracle API version	Directory	Notes
8.0.4	7	oci804_7	Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.
8.0.5	8	oci805_8	Allows access to Oracle 8 and JDBC 2.0 features.
8.1.5	8	OCI815_8	Allows access to Oracle 8 and JDBC 2.0 features.

SGI IRIX

WebLogic jDriver for Oracle for SGI IRIX is available only for the Oracle 7 API. For a list of supported client versions, see [Platform support for JDBC drivers](http://www.weblogic.com/docs51/platforms/index.html#jdbc) at <http://www.weblogic.com/docs51/platforms/index.html#jdbc>.

- Irix users must run Java in N32 mode.
- You must have the Oracle 8.0.5 N32 client installed on your computer.

Add the directory `weblogic/lib/irixsh` to your `LD_LIBRARYN32_PATH`.

Where *weblogic* is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation.

Siemens MIPS

WebLogic jDriver for Oracle for Siemens MIPS is available only for the Oracle 7 API. For a list of supported client versions, see [Platform support for JDBC drivers](http://www.weblogic.com/docs51/platforms/index.html#jdbc) at <http://www.weblogic.com/docs51/platforms/index.html#jdbc>.

Add the directory `weblogic/lib/reliantunix` to your `LD_LIBRARY_PATH`.

Where *weblogic* is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation.

Compaq Tru64 UNIX

WebLogic jDriver for Oracle for Compaq Tru64 UNIX is available only for the Oracle 7 API. For a list of supported client versions, see [Platform support for JDBC drivers](http://www.weblogic.com/docs51/platforms/index.html#jdbc) at <http://www.weblogic.com/docs51/platforms/index.html#jdbc>.

Add the directory `weblogic/lib/tru64unix` to your `LD_LIBRARY_PATH`.

Where *weblogic* is the home directory of your WebLogic Server or WebLogic jDriver for Oracle installation.

Note for Microsoft SDK for Java (Jview) users

- Version 5.00, build 3186 or later of Jview is required for WebLogic jDriver for Oracle.
- Jview does not support JDBC 2.0 features.

Checking connections to the Oracle database

Once you have installed WebLogic jDriver for Oracle you should check that you can use it to connect to your database. A utility called **dbping** is included with WebLogic Server that you can use to test this connection.

To use this helper application, type the following at the command line (on one line):

```
$ java -classpath  
  c:\java\lib\classes.zip;c:\weblogic\classes;  
  c:\weblogic\license;utils.dbping ORACLE user password server
```

Where *c:\weblogic* is the directory containing your WebLogic Server or WebLogic jDriver for Oracle installation and *c:\java* is the path to your JDK.

Note: If you are using Java 2 (JDK 1.2), omit “*c:\java\lib\classes.zip*” from the above command.

For more detailed instructions on how to verify your connection to a DBMS, see [Testing connections](#).

If you have problems, check [Troubleshooting problems with shared libraries on UNIX](http://www.weblogic.com/docs51/techsupport/sharedlibs.html), at <http://www.weblogic.com/docs51/techsupport/sharedlibs.html>.

Setting up a connection pool

If you are using WebLogic jDriver for Oracle with either WebLogic Server or WebLogic Express, you can set up a pool of connections to your Oracle DBMS that will be established when WebLogic Server is started. Since the connections are shared among users, these connection pools eliminate the overhead of opening a new database connection for each user.

Your application then uses a multitier (Type-3) JDBC driver, such as the WebLogic **Pool**, **JTS** or **RMI** driver to connect to WebLogic Server. WebLogic Server then uses WebLogic jDriver for Oracle and one of the existing connections from the pool to connect to the Oracle database on behalf of your application.

Configuring a connection pool with WebLogic Server

1. Include the vendor-supplied native libraries and the WebLogic native libraries for WebLogic jDriver for Oracle in the `PATH` (Windows) or `load library path` (UNIX) of the shell where you will start WebLogic Server (for details, see [Setting your path and client libraries](#) on page 54). For more information on starting WebLogic Server, see [Setting up and Starting WebLogic Server](#) at <http://www.weblogic.com/docs51/install/startserver.html>.
2. Add an entry to the `weblogic.properties` file specifying the connection pool properties (driver name, url, server, password, ACLs, etc.). For example:

```
weblogic.jdbc.connectionPool.OraclePool=\
    url=jdbc:weblogic:oracle:myServer:myPort,\
    driver=weblogic.jdbc.oci.Driver,\
    loginDelaySecs=1,\
    initialCapacity=4,\
    maxCapacity=10,\
    capacityIncrement=2,\
    allowShrinking=true,\
    shrinkPeriodMins=15,\
    refreshMinutes=10,\
    testTable=dual,\
    props=user=myUserName;password=secret;server=myServer;\

weblogic.allow.reserve.weblogic.jdbc.connectionPool.SQLPool=\
    guest,joe,jill
weblogic.allow.reset.weblogic.jdbc.connectionPool.SQLPool=\
    joe,jill
```

```
weblogic.allow.shrink.weblogic.jdbc.connectionPool.SQLPool=\  
joe,jill
```

For more information on setting properties for connection pools, see [JDBC Connection Pools](#), at <http://www.weblogic.com/docs51/admindocs/properties.html#conpools>, in the WebLogic Administrators guide *Setting WebLogic Properties*.

3. Start WebLogic Server.

Using the connection pool in your application

Client-side applications

To use a connection pool in a client-side application, establish the database connection using the WebLogic JDBC/RMI driver. For more information, see

- [Using WebLogic JDBC/RMI and WebLogic Clustered JDBC](#) at http://www.weblogic.com/docs51/classdocs/JDBC_RMI.html.

Server-side applications

To use a connection pool in a server-side application (such as a servlet), establish your database connection using the WebLogic pool or jts drivers. For more information, see:

- [Using connection pools with server-side Java](#) (in *Using WebLogic HTTP Servlets*) at http://www.weblogic.com/docs51/classdocs/API_servlet.html#pools0
- [Creating a startup connection pool](#) at http://www.weblogic.com/docs51/classdocs/API_jdbct3.html#startupconnpool

Using IDEs or debuggers with WebLogic jDrivers

If you are using Symantec Cafe, other IDEs, or debuggers, copy the WebLogic-supplied native library to a new file with a name that ends in `_g` (before the dot).

For example on a Unix machine, copy `libweblogicoci36.so` to `libweblogicoci36_g.so`. For Windows NT, copy `weblogicoci36.dll` to `weblogicoci36_g.dll`

Next step

See Setting your development environment, at <http://www.weblogic.com/docs51/techstart/environment.html> for information on setting up a development environment for running JDBC clients.

Using WebLogic jDriver for Oracle (at http://www.weblogic.com/docs51/classdocs/API_joci.html). Developers Guide for WebLogic jDriver for Oracle.

Index

C

- classes
 - clientclasses 27
 - client-side 27
 - permissions 31
 - server-side 27
 - user 31
 - user-written 27
 - weblogic.policy file 31
- classpath
 - CLASSPATH environment
 - variable 27
 - classpath option 28
 - Java system classpath 28
 - overview 26
 - setting 26
 - setting for NT service 38
 - setting for running under jview
 - 29
 - setting for WebLogic
 - Enterprise Connectivity (WLEC)
 - 43
 - setting in scripts 46
 - setting on command line 40
 - setting to run applications 27
 - setting to run examples 27
 - upgrading 27
 - WebLogic classpath 28
 - weblogic.class.path 28
- clientclasses 27

Cloudscape

- cloudscape.jar 28
- setting classpath for 31

cluster 23

- command line 40
 - classpath 41
 - examples of starting WebLogic Server 42
 - starting WebLogic Server from 39
- container managed persistence 41

D

- dbping 37
- debuggers, using with WebLogic
 - jDriver for Oracle 62

E

- environment, setting 26
- evaluation license 33

H

- hardware requirements 9
- heap memory size 40

I

IDE

- using with WebLogic jDriver

- for Oracle 62
- install 13
 - as Windows NT Service 36
 - from a zip archive 17
 - from InstallShield 13
 - hardware requirements 9
 - JDBC drivers - type 2
 - checking connections 59
 - Oracle 51
 - client libraries 54
 - debuggers 62
 - using an IDE 62
 - using Jview 59
 - software requirements 9
 - UNIX 17
- InstallShield
 - using to install WebLogic Server 13

J

- jar 19
- Java 2
 - See JDK 1.2 42
 - setting up Java security manager 31
- Java Development Kit. *See* JDK
- Java heap memory 40
- Java Runtime Environment *See* JRE
- Java security manager 41
 - and JDK 1.2.1 on UNIX 33
 - setting up 31
 - weblogic.policy 31
- Java system classpath
 - classpath option 28
 - overview 26
- JavaSoft 24
- JDBC drivers - type 4, upgrading 10
- JDBC drivers, installing type 2
 - drivers 51
- JDK

- JavaSoft 24
- SunSoft 24
- JDK (Java Development Kit)
 - installing 24
 - JDK 1.2
 - Java Security Manager 31
- JDK 1.1
 - command line example 42
- JDK 1.2
 - command line example 42
- JDK_HOME setting in scripts 46
- JRE (Java Runtime Environment)
 - limitations for use 25
- Jview
 - See also* Microsoft SDK for Java 29
- jview
 - and class loader 29
 - and dynamic deployment 29
 - command line example 43
 - starting WebLogic Server under 29

K

- keys
 - license 34

L

- license
 - .class file 35
 - adding new keys 34
 - convertLicense utility 36
 - evaluation 33
 - installing 33
 - keys 34
 - adding 34
 - overview 33
 - updating 34
 - upgrading 35

- WeblogicLicense.xml file 34
- XML 34

M

- management
 - WebLogic Console 44
- Microsoft SDK for Java
 - and class loader 29
 - starting WebLogic Server under 29

O

- Oracle
 - install JDBC driver 51

P

- password 45
- PATH, setting 26
- performance pack
 - installing on Solaris 19
 - library path 20

R

- requirements
 - hardware 9
 - software 9

S

- scripts
 - setting JDK_HOME 46
 - startConsole.cmd 44
 - startConsole.sh 44
 - starting WebLogic Console with 44
 - starting WebLogic Server using 44
 - upgrading 9

- security
 - Java security manager 31
- security policy
 - and JDK 1.2.1 on UNIX 33
- security policy file 31
- serverclasses 27
- Setting 54
- setting
 - PATH 26
- setting classpath 26
- software requirements 9
- Solaris
 - performance pack 19
- Start menu
 - configuring parameters to run from 36
 - starting WebLogic Server from 36
 - t3config 36
- starting WebLogic Server
 - clusters 23
 - from the command line 39, 40
 - classpath 41
 - examples 42
 - JDK 1.1 42
 - JDK 1.2 (Java 2) 42
 - jview 43
 - from the Start Menu 36
 - from the WebLogic Console 44
- java heap memory 40
- overview 23
- starting on boot-up 36
- using scripts 44
- WebLogic Enterprise
 - Connectivity 43
- Windows NT 36
 - from the Start Menu 36
 - NT Service 36
 - selecting a JRE 36
 - t3config 36
- startWebLogic.cmd 46

- startWebLogic.sh 46
- startWebLogicJview.cmd 46
- support
 - technical 5
- system
 - PATH 26

T

- t3config 36, 38
- t3console 39
- t3server 39

U

- UNIX
 - group 18
 - install 17
 - installing on 18
 - permissions 18, 20
 - running as web server 20
 - setting library path 20
 - user 18
 - weblogic.properties 19
- unpack
 - jar 19
 - zip 19
- upgrading 9
 - classpath 27
 - EJB 27
 - JDBC type 4 drivers 10
 - license 35
 - scripts 9

V

- version.exe 39

W

- WebLogic classpath 28
- WebLogic COM 30

- WebLogic Console 44
- WebLogic Enterprise Connectivity 43
- WebLogic ZAC 20
- weblogic.class.path 28
 - overview 26
- weblogic.policy file 31
 - and JDK 1.2.1 on UNIX 33
 - modifying 31
 - modifying for third party classes 32
 - modifying for user-written classes 32
- weblogic.properties file
 - modifying 19
- weblogic.system.home 41, 43
- Windows convenience programs 37
 - dbping 37
 - install.exe 38
 - remove.exe 38
 - t3config.exe 38
 - t3console.exe 39
 - t3server.exe 39
 - version.exe 39
- Windows NT
 - NT Service
 - install.exe 38
 - remove.exe 38
 - running multiple instances
 - of WebLogic Server as NT services 37
 - t3config.exe 38
 - t3console.exe 39
 - t3server.exe 39
 - version.exe 39
 - starting WebLogic Server 36
- Windows convenience programs 37

Z

ZAC 20

zip, installing from zip archive 19