BEA WebLogic Enterprise

Installation Guide

WebLogic Enterprise 5.1
Document Edition 5.1
May 2000
Contents

About This Document
What You Need to Know .................................................................................. xvi
e-docs Web Site ................................................................................................. xvi
How to Print the Document .............................................................................. xvii
Related Information .......................................................................................... xvii
Contact Us! ....................................................................................................... xvii
Documentation Conventions ........................................................................... xviii

Part I. Before You Install

1. Preparing to Install the WebLogic Enterprise Software
   Checking the WebLogic Enterprise Product Boxes .......................................... 1-2
   WebLogic Enterprise CDs #1 and #2 Software Components ........................... 1-5
   Hardware and Software Prerequisites for the WebLogic Enterprise Software . 1-7
      For UNIX Systems .................................................................................. 1-8
      For Windows Systems .......................................................................... 1-8
   Overview of Upgrade Considerations ............................................................... 1-9
   Managing Files and Databases ...................................................................... 1-9
      Assigning File Ownership on UNIX Systems........................................ 1-10
      Allocating Disk Space .......................................................................... 1-10
         The WebLogic Enterprise System Disk Management Interface .... 1-10
         Arranging for Raw Disk Space .......................................................... 1-11
         How the WebLogic Enterprise File System Is Organized ............... 1-11
         Space for Queue Spaces (If You Are Using /Q) ............................. 1-13
         Space for Application Servers ......................................................... 1-13
         Space for Stateful Session Bean Storage ...................................... 1-14
      Selecting Directories for the WebLogic Enterprise Files ...................... 1-14
For All Platforms ................................................................. 1-14
For All Server Platforms Supporting the BEA Administration Console ... 1-15
Selecting an Administrative Password ......................................... 1-17
Configuring the WebLogic Enterprise System for Windows 2000 or NT ... 1-18
Configuring the UNIX Operating System for the WebLogic Enterprise
  Software ............................................................................. 1-18
  Semaphores ........................................................................... 1-19
  Message Queues and Messages .............................................. 1-20
  Shared Memory ..................................................................... 1-23
  Other Kernel Tuning Parameters .......................................... 1-24
  Calculating IPC Requirements ............................................ 1-25

Part II. T-Engine Installation

2. WebLogic Enterprise T-Engine Installation on Windows Systems

Platforms Supported ...................................................................... 2-2
If You Are Upgrading from a Previous Release .............................. 2-2
  Starting with a Clean System .................................................. 2-3
Installing the WebLogic Enterprise Software on Microsoft Windows 2000 or NT
  4.0 Systems ........................................................................ 2-7
Preinstallation Considerations .................................................... 2-7
  Backing Up Files .................................................................. 2-7
  Stopping WebLogic Enterprise or BEA Tuxedo Applications and
  Related Services ................................................................... 2-8
  Checking That Your Account Has Administrator Privilege ........... 2-8
Microsoft Windows 2000 or NT 4.0 Installation Procedure ............ 2-8
Setting Microsoft Windows 2000 or NT 4.0 Environment Variables ... 2-25
Installing the Product License After You Install the WebLogic Enterprise
  Software ............................................................................... 2-27
Running Simpapp to Verify the WebLogic Enterprise CORBA ++ Software
  Installation on Microsoft Windows 2000 or NT 4.0 ................. 2-28
Running Simpapp to Verify the WebLogic Enterprise CORBA Java Software
  Installation on Microsoft Windows 2000 or NT 4.0 .................... 2-32
Running a Basic EJB Sample to Verify the WebLogic Enterprise J2EE
  Software Installation .......................................................... 2-35
Installing the WebLogic Enterprise Software on Microsoft Windows 98 and 95 Systems.......................................................... 2-37
Removing (Uninstalling) the WebLogic Enterprise Software from Your System................................................................. 2-43

3. WebLogic Enterprise T-Engine Installation on UNIX Systems
Platforms Supported .................................................................................................................................................. 3-2
Installing the WebLogic Enterprise Software on UNIX Systems .......................................................... 3-3
Preinstallation Considerations ..................................................................................................................... 3-3
Backing Up Files ................................................................................................................................. 3-3
Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services ........................................ 3-4
Checking That Your Account Has the Required Privileges ................................................................. 3-4
UNIX Installation Procedure ..................................................................................................................... 3-4
Installing the T-Engine Product License After You Install the WebLogic Enterprise Software ................. 3-12
If you Installed WebLogic Enterprise on Solaris .................................................................................. 3-13
Running Simpapp to Verify the WebLogic Enterprise CORBA C++ Software Installation .......................................................... 3-13
Running Simpapp to Verify the WebLogic Enterprise CORBA Java Software Installation .......................................................... 3-15
Running a Basic EJB Sample to Verify the WebLogic Enterprise J2EE Software Installation ................. 3-17
Removing (Uninstalling) the WebLogic Enterprise Software from Your System ........................................ 3-19

4. WebLogic Enterprise T-Engine Postinstallation Considerations
Configuring the WebLogic Enterprise System for Microsoft Windows 2000 and NT 4.0................................. 4-2
Accessing the Control Panel Applet ............................................................................................................. 4-3
Accessing Machines on a Network ............................................................................................................. 4-4
Modifying Environment Variables ............................................................................................................. 4-5
Directing WebLogic Enterprise Messages to the Windows Event Log .......................................................... 4-6
Configuring tlisten Processes to Start Automatically ............................................................................. 4-8
Maximizing System Performance ............................................................................................................. 4-9
Setting Up Your Environment on UNIX Systems ........................................... 4-12
Editing a UBBCONFIG File ........................................................................... 4-13
Verifying IPC Requirements .......................................................................... 4-17
Creating the Universal Device List and TLOG ............................................. 4-19
Creating the UDL ...................................................................................... 4-19
Creating the TLOG .................................................................................... 4-20
Starting the tlisten Process on UNIX Systems ............................................. 4-20
Running buildtms and buildXAJS for WebLogic Enterprise Applications That
Use XA Resource Managers ....................................................................... 4-22
Using the TYPE Parameter in the UBBCONFIG File ................................ 4-22
Internet Browser Requirements .................................................................... 4-23

5. BEA Administration Console Startup

System Requirements .................................................................................... 5-1
Platforms Supported ..................................................................................... 5-1
Hardware Requirements .............................................................................. 5-2
Operating System Requirements ............................................................... 5-2
Browser Requirements ............................................................................... 5-3
Setting Up Your Environment ................................................................... 5-3
Starting tuxwsvr ......................................................................................... 5-4
Starting wlisten ............................................................................................ 5-4
Starting the BEA Administration Console ................................................. 5-5

6. WebLogic Enterprise T-Engine Platform Data Sheets

Supported Platforms ..................................................................................... 6-2
Compaq Tru64 UNIX on Alpha Systems ..................................................... 6-3
Available BEA WebLogic Enterprise Version 5.1 Packages ...................... 6-3
Hardware Requirements ............................................................................. 6-4
Software Requirements ............................................................................. 6-4
Network Requirements .............................................................................. 6-6
Disk Space Requirements ........................................................................... 6-7
Mounting and Unmounting the CD ............................................................ 6-8
Tuning Parameters ..................................................................................... 6-8
HP-UX Version 11.0 (32-bit) on HP 9000 Series ......................................... 6-11
WebLogic Enterprise 5.1 Components ..................................................... 6-11
Hardware Requirements ................................................................. 6-12
Software Requirements ................................................................. 6-12
Network Requirements ................................................................. 6-14
Disk Space Requirements .............................................................. 6-14
Mounting and Unmounting the CD ............................................... 6-15
Tuning Parameters ........................................................................ 6-16
IBM AIX 4.3.3 .................................................................................. 6-18
Available BEA WebLogic Enterprise Version 5.1 Packages .......... 6-18
Hardware Requirements ................................................................. 6-19
Software Requirements ................................................................. 6-19
Network Requirements ................................................................. 6-21
Disk Space Requirements .............................................................. 6-22
Mounting and Unmounting the CD ............................................... 6-23
Tuning Parameters ........................................................................ 6-23
Microsoft Windows 2000 and NT 4.0 (SP5) on Intel ....................... 6-24
BEA WebLogic Enterprise Version 5.1 Components ...................... 6-24
Hardware Requirements ................................................................. 6-25
Software Requirements ................................................................. 6-25
Network Requirements ................................................................. 6-28
Disk Space Requirements .............................................................. 6-28
Tuning Parameters ........................................................................ 6-29
Microsoft Windows 98 and 95 ........................................................ 6-29
Available BEA WebLogic Enterprise Version 5.1 Packages .......... 6-29
Hardware Requirements ................................................................. 6-30
Software Requirements ................................................................. 6-30
Network Requirements ................................................................. 6-31
Disk Space Requirements .............................................................. 6-32
SCO UnixWare 7.1.1 ........................................................................ 6-33
Available BEA WebLogic Enterprise Version 5.1 Packages .......... 6-33
Hardware Requirements ................................................................. 6-33
Software Requirements ................................................................. 6-34
Network Requirements ................................................................. 6-35
Disk Space Requirements .............................................................. 6-35
Mounting and Unmounting the CD ............................................... 6-36
Tuning Parameters ........................................................................ 6-37
Sun Microsystems Solaris 2.6 and Solaris 7 (32-Bit) SPARC ....................... 6-39
Available BEA WebLogic Enterprise Version 5.1 Packages .................. 6-39
Hardware Requirements ............................................................................. 6-40
Software Requirements .............................................................................. 6-40
Network Requirements ............................................................................. 6-42
Disk Space Requirements .......................................................................... 6-43
Mounting and Unmounting the CD ......................................................... 6-44
Tuning Parameters .................................................................................... 6-44

Part III. J-Engine Installation

7. Overview of Installing WebLogic Server 5.1
   What’s New in WebLogic Server, Version 5.1 .............................................. 7-1
   Checking Your Package ............................................................................. 7-2
   Hardware Requirements ............................................................................. 7-2
   Software Requirements .............................................................................. 7-3
   Upgrading from an Earlier Release of WebLogic Server ......................... 7-3
      Important Notes .................................................................................. 7-3
      Steps to Take Before Installation ....................................................... 7-4
      Steps to Take After Installation .......................................................... 7-4
   Installing WebLogic Server on Your Platform ........................................... 7-4

8. Installing WebLogic Server Using the InstallShield Distribution (Windows NT)
   Uninstalling a Previous Release ............................................................. 8-2
   Running the InstallShield Program ........................................................ 8-2
   Next Steps ............................................................................................... 8-3

9. Installing WebLogic Server from a Zip Archive (UNIX, Windows NT)
   Overview ................................................................................................. 9-1
      Installing on Windows NT Using the Zip Archive .................................. 9-1
      Upgrading from a Previous Release ....................................................... 9-2
      Installing from the Zip Archive ............................................................ 9-2
   Next Steps ............................................................................................... 9-5
10. Setting Up and Starting WebLogic Server 5.1

Installing a JDK ............................................................................................... 10-2
  Native Versus Green Threads ................................................................... 10-3
  Hot Spot.................................................................................................... 10-3
Setting the Environment .................................................................................. 10-4
  Setting the System PATH ........................................................................ 10-4
  Setting the Classpath ............................................................................... 10-4
    Upgrading from a Previous Release.................................................. 10-5
    Setting your Java System Classpath............................................ 10-6
    Setting your WebLogic Classpath................................................. 10-6
    Microsoft SDK for Java (JView) .................................................. 10-7
Starting WebLogic Server Statically ........................................................ 10-8
Cloudscape DBMS ................................................................................... 10-9
  Setting up the Java Security Manager for Java 2 ..................................... 10-9
    Modifying the weblogic.policy File for General Use ................. 10-9
    Modifying the weblogic.policy File for Third-party or User-written
    Classes ..................................................................................... 10-10
Installing a WebLogic License............................................................... 10-11
  Evaluation Licenses ........................................................................ 10-11
  Other Licenses................................................................................. 10-12
  Updating a License........................................................................ 10-12
  Upgrading Licenses from a Previous Release................................. 10-13
Starting WebLogic Server on Windows NT ................................................. 10-14
  Start Menu .............................................................................................. 10-14
  NT Service.............................................................................................. 10-14
  Windows Convenience Programs .......................................................... 10-15
Starting WebLogic Server from the Command Line .................................... 10-18
  Important Note Regarding WebLogic RMI over IIOP .................... 10-18
Requirements for Starting WebLogic Server ......................................... 10-18
Command-line Examples ....................................................................... 10-20
  JDK 1.1.x Example ......................................................................... 10-20
  JDK 1.2 (Java 2) Example .............................................................. 10-20
  Jview Example ........................................................................... 10-21
  Additional Options ........................................................................ 10-21
  Starting WebLogic Enterprise Connectivity .................................... 10-21
Starting WebLogic Server from the WebLogic Console............................... 10-22
Starting WebLogic Server Using Scripts....................................................... 10-24
Next Steps.................................................................................................... 10-24
   Set Up Your Development Environment .............................................. 10-24
   Install JDBC Drivers for Use with WebLogic Server ......................... 10-24
      Oracle .......................................................................................... 10-25
      Informix .................................................................................... 10-25
      Microsoft SQL Server ............................................................... 10-25
      Sybase ..................................................................................... 10-25
Other Documentation .................................................................................. 10-25

11. Installing WebLogic jDriver for Oracle

   Overview ............................................................................................. 11-1
   Other JDBC Drivers ............................................................................. 11-2
   Installation Steps .................................................................................. 11-3
      Editing an Entry to the XML License File ........................................ 11-5
   Setting Your Path and Client Libraries .............................................. 11-6
      JDBC 2.0 ...................................................................................... 11-7
      Platform Considerations ................................................................ 11-7
      Directory to Put in Your System PATH ........................................... 11-7
         Windows NT .............................................................................. 11-7
         Solaris ..................................................................................... 11-8
         IBM AIX ................................................................................... 11-9
         HP-UX 11 ............................................................................... 11-9
         SGI IRIX ............................................................................... 11-9
         Siemens MIPS ......................................................................... 11-10
         Compaq Tru64 UNIX ............................................................... 11-10
   Note for Microsoft SDK for Java (Jview) Users .................................... 11-10
   Checking Connections to the Oracle Database ................................... 11-11
      Setting Up a Connection Pool ....................................................... 11-11
      Configuring a Connection Pool with WebLogic Server .............. 11-12
      Using the Connection Pool in Your Application ......................... 11-13
   Using IDEs or Debuggers with WebLogic jDrivers .......................... 11-13
   Next Step .......................................................................................... 11-14
Part IV. Encryption Package Installation

12. WebLogic Enterprise Encryption Package Installation on Windows Systems

Before You Install ................................................................. 12-2
   Installing a WebLogic Enterprise 5.1 Encryption Package Product to Support
     WebLogic Enterprise Connectivity ........................................... 12-3

   Confirming That the WebLogic Enterprise 5.1 Software Has Been Installed . 12-3
   LDAP Information Required During the Installation ....................... 12-4
   Before Re-installation, Backup LDAP Files ............................... 12-5
   Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related
     Services .................................................................................. 12-5
   Check That Your Account Has Administrator Privileges ................ 12-6

Platforms Supported ................................................................................ 12-6
Installing the WebLogic Enterprise 5.1 Encryption Package on Microsoft
   Windows Systems .................................................................. 12-7
   Microsoft Windows Installation Procedure .............................. 12-7

Removing (Uninstalling) the WebLogic Enterprise Encryption Package from
   Your System .............................................................................. 12-14

13. WebLogic Enterprise Encryption Package Installation on UNIX Systems

Before You Install ................................................................. 13-2
   Installing a WebLogic Enterprise 5.1 Encryption Package Product to Support
     WebLogic Enterprise Connectivity ........................................... 13-3

   Confirming That the WebLogic Enterprise 5.1 Software Has Been
     Installed ............................................................................... 13-3
   Environment Variables ............................................................ 13-4
   LDAP Information Required During the Installation ....................... 13-4
   Before Re-installation, Back Up LDAP Files ............................... 13-5
   Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related
     Services .................................................................................. 13-5
   Checking That Your Account Has Administrator Privileges ............ 13-6

Platforms Supported ................................................................................ 13-6
Installing WebLogic Enterprise Encryption Package on UNIX Systems .... 13-7
Part V. BEA Jolt Installation

14. Preparing to Install BEA Jolt

What’s New in Jolt ................................................................. 14-1
What’s Changed in Jolt .......................................................... 14-3
System Requirements ......................................................... 14-3
  WebLogic Enterprise ......................................................... 14-4
  Web Servers Supported ................................................... 14-4
  Supported Platforms for Jolt Server ............................... 14-4
  Client Support .................................................................. 14-6
  Jolt Client Requirements ................................................ 14-7
  Jolt Client Class Library .................................................. 14-8
Release Migration/Interoperability ............................... 14-8
Jolt ASP Connectivity for BEA Tuxedo ......................... 14-9
  Requirements ................................................................. 14-9
  Installation Instructions .................................................. 14-9
Preinstallation Checklist .................................................... 14-12

15. Installing BEA Jolt

Microsoft Windows NT Installation Instructions .............. 15-1
UNIX System Installation Instructions ............................. 15-11
  Invoking the UNIX Installation Script ......................... 15-12
  Unix/Linux System Installation Script ......................... 15-12
Licensing Jolt for WebLogic Enterprise 5.1 ..................... 15-16
  UNIX Licensing Instructions ....................................... 15-17
  NT Licensing Instructions ........................................... 15-17

16. Configuring the BEA Jolt System

Quick Configuration .......................................................... 16-2
  Configure Jolt on BEA Tuxedo ....................................... 16-2
  Edit the UBBCONFIG file ............................................. 16-2
  Configure the Jolt Repository ...................................... 16-3
17. Post Installation Considerations for BEA Jolt

Installing JRLY After Normal Installation .............................................. 17-1
Installing JRLY on UNIX ........................................................................ 17-1
Installing JRLY on NT .......................................................................... 17-2
Uninstalling Jolt ...................................................................................... 17-2
Uninstalling Jolt From NT ..................................................................... 17-2
Uninstalling Jolt From UNIX ................................................................. 17-4

Index
About This Document

This document explains how to install the BEA WebLogic Enterprise™ T-Engine software, J-Engine software, Encryption Package software, and BEA Jolt™ software. The topics are organized as follows:

- Part I, “Before You Install,” which contains the following topic:
  - Chapter 1, “Preparing to Install the WebLogic Enterprise Software”

- Part II, “T-Engine Installation,” which includes the following topics:
  - Chapter 2, “WebLogic Enterprise T-Engine Installation on Windows Systems”
  - Chapter 3, “WebLogic Enterprise T-Engine Installation on UNIX Systems”
  - Chapter 5, “BEA Administration Console Startup”
  - Chapter 4, “WebLogic Enterprise T-Engine Postinstallation Considerations”
  - Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets”

- Part III, “J-Engine Installation,” which includes the following topics:
  - Chapter 7, “Overview of Installing WebLogic Server 5.1”
  - Chapter 8, “Installing WebLogic Server Using the InstallShield Distribution (Windows NT)”
  - Chapter 9, “Installing WebLogic Server from a Zip Archive (UNIX, Windows NT)”
  - Chapter 10, “Setting Up and Starting WebLogic Server 5.1”
  - Chapter 11, “Installing WebLogic jDriver for Oracle”

- Part IV, “Encryption Package Installation,” which includes the following topics:
What You Need to Know

This document is intended mainly for system administrators and installers who will install one or more WebLogic Enterprise server, client, or administration components.

e-docs Web Site

The BEA WebLogic Enterprise product documentation is available from the BEA Systems, Inc. corporate Web site. From the BEA Home page, click the Product Documentation button or go directly to the “e-docs” Product Documentation page at http://e-docs.bea.com.
How to Print the Document

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

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

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

Related Information

Before installing the BEA WebLogic Enterprise software, read the BEA WebLogic Enterprise Release Notes.

For more information about topics covering CORBA, Java 2 Enterprise Edition (J2EE), BEA Tuxedo®, distributed object computing, transaction processing, C++ programming, and Java programming, see the WebLogic Enterprise Bibliography in the WebLogic Enterprise online documentation.

Contact Us!

Your feedback on the BEA WebLogic Enterprise documentation is important to us. Send us e-mail at docsupport@bea.com if you have questions or comments. Your comments will be reviewed directly by the BEA Systems, Inc. professionals who create and update the WebLogic Enterprise documentation.
In your e-mail message, please indicate that you are using the documentation for the BEA WebLogic Enterprise 5.1 release.

If you have any questions about this version of BEA WebLogic Enterprise, or if you have problems installing and running BEA WebLogic Enterprise, contact BEA Customer Support through BEA WebSUPPORT at [www.bea.com](http://www.bea.com). You can also contact Customer Support by using the contact information provided on the Customer Support Card, which is included in the product package.

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

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

**Documentation Conventions**

The following documentation conventions are used throughout this document.

<table>
<thead>
<tr>
<th>Convention</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>boldface text</strong></td>
<td>Indicates terms defined in the glossary.</td>
</tr>
<tr>
<td>Ctrl+Tab</td>
<td>Indicates that you must press two or more keys simultaneously.</td>
</tr>
<tr>
<td><em>italics</em></td>
<td>Indicates emphasis or book titles.</td>
</tr>
</tbody>
</table>
## Documentation Conventions

<table>
<thead>
<tr>
<th>Convention</th>
<th>Item</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>monospace text</strong></td>
<td>Indicates code samples, commands and their options, data structures and their members, data types, directories, and filenames and their extensions. Monospace text also indicates text that you must enter from the keyboard.</td>
</tr>
<tr>
<td><strong>Examples:</strong></td>
<td></td>
</tr>
<tr>
<td><code>#include &lt;iostream.h&gt;</code> void main ( ) the pointer psz chmod u+w * \tux\data\ap .doc tux.doc BITMAP float</td>
<td></td>
</tr>
<tr>
<td><strong>monospace boldface text</strong></td>
<td>Identifies significant words in code.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>void commit ()</td>
<td></td>
</tr>
<tr>
<td><strong>monospace italic text</strong></td>
<td>Identifies variables in code.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>String expr</td>
<td></td>
</tr>
<tr>
<td><strong>UPPERCASE TEXT</strong></td>
<td>Indicates device names, environment variables, and logical operators.</td>
</tr>
<tr>
<td><strong>Examples:</strong></td>
<td></td>
</tr>
<tr>
<td>LPT1 SIGNON OR</td>
<td></td>
</tr>
<tr>
<td>( )</td>
<td>Indicates a set of choices in a syntax line. The braces themselves should never be typed.</td>
</tr>
<tr>
<td>[ ]</td>
<td>Indicates optional items in a syntax line. The brackets themselves should never be typed.</td>
</tr>
<tr>
<td><strong>Example:</strong></td>
<td></td>
</tr>
<tr>
<td>buildobjclient [-v] [-o name ] [-f file-list]... [-l file-list]...</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Separates mutually exclusive choices in a syntax line. The symbol itself should never be typed.</td>
</tr>
</tbody>
</table>
### Convention | Item
--- | ---
... | Indicates one of the following in a command line:
- 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.

Example:
```
buildobjcclient [-v] [-o name ] [-f file-list]...
```

. | Indicates the omission of items from a code example or from a syntax line.
. | The vertical ellipsis itself should never be typed.
Part I  Before You Install

Chapter 1.  Preparing to Install the WebLogic Enterprise Software
BEA WebLogic Enterprise is a sophisticated software product. It should not be installed without proper planning.

This topic includes the following sections:

- Checking the WebLogic Enterprise Product Boxes
- WebLogic Enterprise CDs #1 and #2 Software Components
- Hardware and Software Prerequisites for the WebLogic Enterprise Software
- Overview of Upgrade Considerations
- Managing Files and Databases
- Selecting Directories for the WebLogic Enterprise Files
- Selecting an Administrative Password
- Configuring the WebLogic Enterprise System for Windows 2000 or NT
- Configuring the UNIX Operating System for the WebLogic Enterprise Software
Checking the WebLogic Enterprise Product

Boxes

The WebLogic Enterprise software comes in two boxes:

- The smaller box contains the following four compact discs (CDs):
  
  - CDs #1 and #2, which contain the WebLogic Enterprise “T-Engine” software components. These components enable you to build scalable applications based on the following technologies: CORBA C++, CORBA Java, J2EE (EJBeans, RMI, JNDI, JDBC), and BEA Tuxedo.
  
  CD #1 contains the T-Engine technologies for installation on Microsoft Windows 2000 systems, Windows NT systems, HP-UX systems, or Sun Solaris 2.6 or Solaris 7 systems (standard mode or compatibility mode). Using CD #1, you can also install WebLogic Enterprise T-Engine client components (only) on Windows 98 or Windows 95 systems.

  CD #2 contains the T-Engine technologies for installation on Compaq Tru64 systems, or IBM AIX systems, or SCO Unixware systems.

  - CD #3, which contains the WebLogic Enterprise “J-Engine” software components and BEA Jolt™.

  The J-Engine software enables you to build scalable applications based on J2EE technologies. The features include Web application servers, servlets, Java Server Pages, EJBeans, and connectivity to WebLogic Enterprise T-Engine applications via the WebLogic Enterprise Connectivity (sometimes called WLEC) product.

  BEA Jolt is a Java-based interface to the BEA Tuxedo system that extends the functionality of existing Tuxedo applications to include Intranet- and Internet-wide availability.

  - The WebLogic Enterprise 5.1 Online Documentation CD, which documents the T-Engine features, the J-Engine features, and the Jolt features.

- The larger box contains the following printed documents and diskettes:
  
  - BEA WebLogic Enterprise Release Notes
  
  - BEA WebLogic Enterprise Installation Guide (this document)
Checking the WebLogic Enterprise Product Boxes

- BEA WebLogic Enterprise Getting Started
- BEA WebLogic Server Introduction
- The quick reference Customer Support Agreement
- BEA software license agreement
- Two separate 3.5-inch diskettes that contain the product licenses for the WebLogic Enterprise 5.1 software, as summarized in the following table:

<table>
<thead>
<tr>
<th>If you have purchased....</th>
<th>You have a license to use...</th>
</tr>
</thead>
</table>
| WebLogic Enterprise CORBA and ATMI (SDK or run-time license) | - T-Engine CORBA C++ and Java, and ATMI software  
- J-Engine software for 30 days  
- BEA Jolt software for 30 days |
| WebLogic Enterprise CORBA, ATMI, and J2EE (SDK or run-time license) | - T-Engine CORBA C++ and Java, and ATMI software  
- T-Engine J2EE software  
- J-Engine software |

Refer to the BEA WebLogic Enterprise Release Notes for information about the distribution mechanism for the T-Engine, J-Engine, and Jolt licenses.

If you ordered the WebLogic Enterprise 5.1 56-bit or 128-bit Encryption Package software product, an additional, separate package contains the CD for this software. Installation of the optional Encryption Package software on Windows systems is explained in Chapter 12, “WebLogic Enterprise Encryption Package Installation on Windows Systems.” Installation of the optional Encryption Package software on UNIX systems is explained in Chapter 13, “WebLogic Enterprise Encryption Package Installation on UNIX Systems.”

Figure 1-1 shows the contents of the four CDs that come in the WebLogic Enterprise product box, plus the optional Encryption Package CDs that are separately orderable.
1 Preparing to Install the WebLogic Enterprise Software

Figure 1-1 Distribution CDs for WebLogic Enterprise 5.1

<table>
<thead>
<tr>
<th>CD # 1</th>
<th>CD # 2</th>
<th>CD # 3</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image1" alt="CD" /></td>
<td><img src="image2" alt="CD" /></td>
<td><img src="image3" alt="CD" /></td>
</tr>
<tr>
<td>T-Engine software for HP-UX, Windows NT, Windows 2000, Solaris (standard &amp; compatibility modes)</td>
<td>T-Engine software for IBM AIX, Compaq Tru64 UNIX, SCO UnixWare</td>
<td>J-Engine software and BEA Jolt software</td>
</tr>
</tbody>
</table>

Separately Orderable:

- Online Documentation CD for all T-Engine, J-Engine, and BEA Jolt features
- 56-bit or 128-bit Encryption Package Software CD

**Note:** Using CD #1, you can also install WebLogic Enterprise T-Engine client components on Windows 98 or Windows 95 systems.
WebLogic Enterprise CDs #1 and #2
Software Components

The WebLogic Enterprise CDs #1 and #2 contain the following software components.

- The following WebLogic Enterprise server components:
  - CORBA C++
  - CORBA Java
  - J2EE
  - BEA Tuxedo 6.5 for WebLogic Enterprise 5.1

- The following WebLogic Enterprise client components:
  - CORBA C++ client Object Request Broker (ORB), including environmental objects
  - CORBA Java client Object Request Broker (ORB), including environmental objects
  - RMI/EJB client
  - ActiveX client for Windows systems, including the BEA Application Builder graphical user interface
  - BEA Tuxedo 6.5 client

- BEA Administration Console

For a list of the platforms supported for this release of the WebLogic Enterprise software, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”

Notes: The installation procedure for the J-Engine software components on CD #3 is described in Part III, J-Engine Installation. The installation procedure for the BEA Jolt software components on CD #3 is described in Part V, BEA Jolt Installation. The optional BEA WebLogic Enterprise 56-bit or 128-bit Encryption Package software, which provides Secure Sockets Layer (SSL) and Link-Level Encryption (LLE) for WebLogic Enterprise applications, is
Preparing to Install the WebLogic Enterprise Software

shipped to you only if you purchased this software. It is packaged and
distributed on a separate CD. The installation procedure for this optional
software is described in Part IV, Encryption Package Installation.

The WebLogic Enterprise 5.1 T-Engine installation procedure lets you select or
deselect the T-Engine components that you want to install. You can also select or
deselect specific subcomponents within the servers or clients categories.

At least one component or subcomponent must be selected for installation. Selecting a
main component category causes all of its subcomponents to be selected. Deselecting
a component causes all of its subcomponents to be deselected. Deselecting all
subcomponents causes their parent component to be deselected.

The main component categories are:

- Servers
- Clients
- Administration Console

Within the Servers category, the options are:

- BEA Tuxedo Servers
- BEA CORBA C++ Servers
- BEA CORBA Java Servers
- BEA J2EE Servers

This feature allows you to install one or more server components on the target system.

Please note:

- The BEA Tuxedo server software is always installed as a base component for
  any of the other WebLogic Enterprise servers.
- The CORBA Java and J2EE server components are always installed together, even if you only select one of those items.
- When you install an individual type of server software, its corresponding client
  software is also installed. For example, if you install the J2EE Servers software,
  the RMI/EJB client software is also installed automatically. Therefore, if you
  have installed a type of server software, you do not need to perform a separate
  installation for the client software for that server.
If you select the Clients component, you can indicate which types of clients you want to install. The Client options are:

- BEA Tuxedo Client
- BEA CORBA C++ Client
- BEA CORBA Java Client
- BEA RMI/EJB Client
- BEA ActiveX Client (Windows systems only)

The Administration category consists of the Administration Console and does not have any subcomponents. For information about how to start this console after it is installed, refer to Chapter 5, “BEA Administration Console Startup.” For information about how to use this console, refer to the online help that is accessible through the Console’s Help button.

Hardware and Software Prerequisites for the WebLogic Enterprise Software

The WebLogic Enterprise software must be installed on each machine that will run a WebLogic Enterprise client or server application.

**Note:** Do not share the WebLogic Enterprise executables across remote file systems.

The BEA Administration Console must be installed in a file system that supports long filenames (that is, those containing more than 14 characters).

Before you can install the optional WebLogic Enterprise Encryption Package 5.1 software, you must first install at least one WebLogic Enterprise 5.1 server component, or at least one of the following WebLogic Enterprise 5.1 client component options:

- All WebLogic Enterprise client components (recommended)
- BEA C++ client
- BEA Tuxedo client
If you are installing the WebLogic Enterprise 5.1 Encryption Package software on a Microsoft Windows 98 or Windows 95 client system, you must first install at least one of the WebLogic Enterprise 5.1 client components shown in the previous list. Installation of the optional Encryption Package software on Windows systems is explained in Chapter 12, “WebLogic Enterprise Encryption Package Installation on Windows Systems.” Installation of the optional Encryption Package software on UNIX systems is explained in Chapter 13, “WebLogic Enterprise Encryption Package Installation on UNIX Systems.”

For details about the hardware and software prerequisites for all platforms on which the WebLogic Enterprise software is supported, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.” Check the data sheet for each platform on which you plan to install the WebLogic Enterprise software.

For UNIX Systems

You need the following information and resources before installing the WebLogic Enterprise software on a UNIX system:

- A system that meets the hardware and software requirements described in Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
- The superuser password so that you can mount the CD as a file system.
- The name of a file system with enough free space for the WebLogic Enterprise software packages you want to install. For disk space requirements, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”

For Windows Systems

You need the following resources before installing the WebLogic Enterprise software on a Microsoft Windows system:

- Administrative privileges.
- A system that meets the hardware and software requirements described in Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
Overview of Upgrade Considerations

- Enough disk space for the packages you want to install. For disk space requirements, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”

Note: Microsoft Windows 98 and 95 systems support only the WebLogic Enterprise Client software. Microsoft Windows 98 and 95 systems do not support the full WebLogic Enterprise system software (server and client) or the BEA Administration Console software.

Overview of Upgrade Considerations

If you are installing the WebLogic Enterprise 5.1 software on a Microsoft Windows 2000, NT, 98, or 95 system that contains a previous version of WebLogic Enterprise, M3, or BEA Tuxedo software, there are important upgrade considerations. In general, BEA recommends that you use the Windows Add/Remove (uninstall) program to remove the previous WebLogic Enterprise, M3, or BEA Tuxedo software on the target system, before you install WebLogic Enterprise 5.1. These considerations are discussed in Chapter 2, “WebLogic Enterprise T-Engine Installation on Windows Systems.”

Managing Files and Databases

This section explains how to assign ownership of the WebLogic Enterprise system files to the system administrator, and how to set up your disk to accommodate those files.
Assigning File Ownership on UNIX Systems

If you are installing the WebLogic Enterprise software on a UNIX system, BEA recommends that you create a separate user account for the WebLogic Enterprise system administrator and give ownership of the WebLogic Enterprise system files to that account.

Allocating Disk Space

A running WebLogic Enterprise client or server application needs disk space for system files and for the application’s database(s). You do not use this space until you begin to develop or run your WebLogic Enterprise client or server application, but it is important to plan for this space before installing the software. To help explain what is involved, the following sections describe how the WebLogic Enterprise system handles files.

For more information about the commands discussed in this section, see the following documents:

- The Administration topics in the WebLogic Enterprise online documentation
- The BEA Tuxedo Reference Manual, which is included in the WebLogic Enterprise online documentation

The WebLogic Enterprise System Disk Management Interface

The WebLogic Enterprise system has a facility, the Disk Management Interface (DMI), that manages logical files within a single disk device or set of devices. Among other things, the DMI stores binary configuration tables and the transaction log.

The WebLogic Enterprise disk management software supports the notion of a WebLogic Enterprise file system that is distinct from any operating system file system. (For the remainder of this discussion, the term OS file system is used to refer to any operating system file system.)

Administrative access to the DMI to create, initialize, or destroy entries in the WebLogic Enterprise file system is through tmadmin administrative commands.

There are two ways to physically store the logical files managed by the DMI:
Managing Files and Databases

- Physical storage can be on an OS file system.
- Disk space outside the control of all OS file systems can be set aside for the WebLogic Enterprise system.

Files reside on special device files in that disk space, and the DMI manages the files directly. Space outside the OS file system is usually referred to as raw disk space. Not only is I/O faster when done by system calls reading directly from and writing directly to device special files on raw disks, raw disk space is preferred when it is important to know for certain that a physical write() has been done.

With the OS file system, the precise moment at which a write() is done cannot be relied upon. In the WebLogic Enterprise system, accurate control of the write operation is particularly important for entries in the transaction log. With multiple users, control of the write operation is also an important element in assuring database consistency.

Arranging for Raw Disk Space

If you decide to use raw disk space for your WebLogic Enterprise client or server application, you may find that the hard disk devices on your machine are fully allocated to file systems such as / (root), /usr, and other UNIX file systems. If that is the case, it is necessary to repartition your hard disk device to set aside some partitions that are not to be used for an OS file system. Information about how to do this can be found in the system administration documentation for your particular platform.

Notes: Repartitioning disks can render the machine unusable and should be attempted only by experienced UNIX system administrators.

On Microsoft Windows NT platforms, the default behavior is unbuffered I/O; no special arrangements are needed.

How the WebLogic Enterprise File System Is Organized

A WebLogic Enterprise file system has a Volume Table of Contents (VTOC) that lists files on a set of devices named in the Universal Device List (UDL). The UDL contains information about the location of the physical storage space for the WebLogic Enterprise tables.
In a WebLogic Enterprise system, all the system files might be stored together on the same raw disk slice or OS file system file. While it is possible to use regular OS file system files for the configuration tables, it is strongly recommended that the transaction log (TLOG) be stored on a raw disk device.

Because the TLOG seldom needs to be larger than 100 blocks and because disk partitions are always substantially larger than 100 blocks, it may make sense to use the same device for everything. The pathname of the device needs to be contained in both the TUXCONFIG and the FSCONFIG environment variables.

Listing 1-1 shows approximately how the content might appear.

Listing 1-1  VTOC and UDL Output

Output based on setting FSCONFIG=$TUXCONFIG, and invoking tmadmin:

No bulletin board exists. Entering boot mode.

> livtoc
Volume Table of Contents on /usr2/bank/tuxconfig:
0: VTOC: Device 0 Offset 0 Pages 7
1: UDL: Device 0 Offset 7 Pages 28
2: _RESOURCE_SECT: Device 0 Offset 35 Pages 3
3: _MACHINES_SECT: Device 0 Offset 38 Pages 40
4: _GROUPS_SECT: Device 0 Offset 78 Pages 40
5: _SERVERS_SECT: Device 0 Offset 118 Pages 40
6: _SERVICES_SECT: Device 0 Offset 158 Pages 20
7: _ROUTING_SECT: Device 0 Offset 178 Pages 100
8: _NETWORK_SECT: Device 0 Offset 278 Pages 20
9: _MIBPERMS_SECT: Device 0 Offset 298 Pages 2

# If the TLOG is stored on the same device, there will be an # entry something like:

9: TLOG1: Device 0 Offset 236 Pages 100
> q
The WebLogic Enterprise system administrator must ensure that raw disk slices are available, as needed, on each machine participating in a WebLogic Enterprise domain. The size of entities in the WebLogic Enterprise file system are shown in Table 1-1.

### Table 1-1 Size of System Tables

<table>
<thead>
<tr>
<th>Entity</th>
<th>512-byte Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>VTOC</td>
<td>1</td>
</tr>
<tr>
<td>TUXCONFIG</td>
<td>270</td>
</tr>
<tr>
<td>TLOG</td>
<td>100 (default)</td>
</tr>
<tr>
<td>UDL</td>
<td>28</td>
</tr>
<tr>
<td>TOTAL</td>
<td>399</td>
</tr>
</tbody>
</table>

The size of the TUXCONFIG file is larger if there are more entries in the configuration file (UBBCONFIG). The administrator is encouraged to allocate additional space for dynamic reconfiguration and for growth of the application. The default size assumed by the crdl subcommand of tmadmin is 1000 blocks, which should be adequate for the initial installation.

**Space for Queue Spaces (If You Are Using /Q)**

If your WebLogic Enterprise application is using the BEA Tuxedo system/Q for store-and-forward queue management, your queue space can be listed in the same UDL and can be managed by the WebLogic Enterprise VTOC.

**Space for Application Servers**

As you are calculating the space requirements for the WebLogic Enterprise system, also consider the requirements of the server machines that perform the work of the server application. These requirements are specified by the application, and they are in addition to the requirements for the WebLogic Enterprise system itself (unless otherwise specified).
Space for Stateful Session Bean Storage

When you calculate the space requirements for the WebLogic Enterprise system, also consider the requirements for saving the state of EJB Stateful Session Beans. These requirements are specified by the application, and they are in addition to the requirements for the WebLogic Enterprise system itself.

Selecting Directories for the WebLogic Enterprise Files

During the installation process, you are prompted to make decisions about where, in your file system, a number of your WebLogic Enterprise directories and files are installed. To help you plan for this part of the process, this section describes the directories and files about which you are prompted to make a decision, as follows:

- “For All Platforms” should be read by anyone installing the WebLogic Enterprise software.
- “For All Server Platforms Supporting the BEA Administration Console” should be read by anyone installing the BEA Administration Console for WebLogic Enterprise administration.

For All Platforms

You are prompted for a pathname for the base directory of your WebLogic Enterprise software. This directory must meet the following requirements:

- The directory must be dedicated to the WebLogic Enterprise software. It must not contain files for any other applications.
- The directory must have read, write, and search (execute) permissions for the WebLogic Enterprise administrator.
Selecting Directories for the WebLogic Enterprise Files

In the WebLogic Enterprise documentation, this directory is referred to as $TUXDIR (UNIX systems) or $TUXDIR\ (Windows NT systems), except in cases where a sample path is shown, such as c:\wledir.

For All Server Platforms Supporting the BEA Administration Console

If you are installing the WebLogic Enterprise Administration software, you are prompted to accept or replace the default pathnames and filenames used for the BEA Administration Console components. These default pathnames and filenames are based on the value of $TUXDIR\ (Windows NT systems) or $TUXDIR (UNIX systems) that you specify.

If you are running a commercial Web server, you may find the default settings inappropriate, especially if your server is handling requests from both the BEA Administration Console and other Web programs on the same port. To accommodate this situation, the WebLogic Enterprise software enables you to choose between accepting the defaults and assigning your own pathnames and filenames. The remainder of this section describes the choices you are given, as follows:

1. A pathname for the HTML files—by default, the following HTML files are installed in the directory $TUXDIR\udataobj\webgui (Windows NT systems), and $TUXDIR/udataobj/webgui (UNIX systems). You are prompted to supply your own paths for these files if you prefer to have them installed elsewhere.
   - An HTML template file (webgui.html) that is used by tuxadm as the basis for many screens displayed during a BEA Administration Console session.
   - An HTML file (webguitop.html) that displays legal notices and warnings when the BEA Administration Console is first displayed on the screen.
   - The HTML files that make up the BEA Administration Console documentation. These HTML files are installed in $TUXDIR\help (Windows NT systems) and in $TUXDIR/help (UNIX systems).

Exception: If you are installing the WebLogic Enterprise software on a Microsoft Windows NT platform and the installation program detects an existing Web server, a default directory appropriate for that Web server is used, instead.
Preparing to Install the WebLogic Enterprise Software

2. A pathname for the Java and image files—by default, the class files for the Java applet are installed in one of the following directories. You are prompted to supply your own paths for these files if you prefer to have them installed elsewhere.
   - $TUXDIR\udatabobj\webgui\java (Windows 2000 or NT systems) and $TUXDIR/udatabobj/webgui/java (UNIX systems)
   - A subdirectory called java in the HTML directory you specified after the prompt described in step 1

3. A directory pathname for the CGI program (tuxadm)—specify one of the following (unless the following exception applies):
   - $TUXDIR\udatabobj\webgui\cgi-bin (Windows 2000 OR NT systems)
   - $TUXDIR/udatabobj/webgui/cgi-bin (UNIX systems)
   - A subdirectory called cgi-bin in the HTML directory you specified after the prompt described in step 1

   Exception: If the installation program detects the Microsoft Internet Information Server (IIS) in a standard directory, tuxadm is installed in a subdirectory called scripts in the directory you specified in step 1 as the pathname for the HTML files.

   **Note:** Do not specify $TUXDIR/bin (UNIX systems) or $TUXDIR\bin (Windows 2000 or NT systems) as your CGI directory. If you do, you risk having other WebLogic Enterprise client or server applications executed accidentally by an uninformed user of the BEA Administration Console. You may also introduce a security risk.

4. An alias for the directory pathname for tuxadm. This is the path for the directory in which Web clients expect to find tuxadm. The default is either cgi-bin or /scripts (for UNIX systems) or \cgi-bin or \scripts (for Windows 2000 or NT systems).
Selecting an Administrative Password

The WebLogic Enterprise system uses an administrative password to protect the machine on which it is installed from unauthorized administrative requests and operations (such as tmboot). Whenever administrative communications arrive on this machine through the tlisten and wlisten processes, the WebLogic Enterprise system authenticates the communications by means of the password.

You assign an administrative password during the installation process (to the machine on which the WebLogic Enterprise software is being installed) by entering the password of your choice after the appropriate prompt. The password must be a string of alphanumeric characters in clear-text format. It may contain no more than 80 characters.

A common password is required for two machines in a WebLogic Enterprise domain to communicate successfully. For this reason, you must use the same password whenever you install the WebLogic Enterprise software on multiple machines for a single domain. As described previously, you are prompted to provide the password during the WebLogic Enterprise installation process. If, however, you use a different password for one machine, you must add that password to the tlisten.pw file on each existing machine with which you want that machine to communicate.

For these reasons, you may have more than one administrative password in your tlisten.pw file. A single password file may contain no more than 20 passwords, with one password per line.

The administrative password that you enter during installation is collected by the installation script and is stored in:

`$TUXDIR/udataobj/tlisten.pw` (UNIX systems)
`%TUXDIR%\udataobj\tlisten.pw` (Windows 2000 or NT systems)

Make sure the permissions on your tlisten.pw file are set such that only the WebLogic Enterprise system administrator can read the file.
Preparing to Install the WebLogic Enterprise Software

Configuring the WebLogic Enterprise System for Windows 2000 or NT

You cannot configure your WebLogic Enterprise system for Microsoft Windows 2000 or NT until after you install the WebLogic Enterprise software and license. After you complete the installation as described in Chapter 2, “WebLogic Enterprise T-Engine Installation on Windows Systems,” refer to the section “Configuring the WebLogic Enterprise System for Microsoft Windows 2000 and NT 4.0” on page 4-2 for instructions on configuring the WebLogic Enterprise system for Windows 2000 or NT.

Configuring the UNIX Operating System for the WebLogic Enterprise Software

The WebLogic Enterprise software uses the UNIX operating system Interprocess Communications (IPC) resources.

IPC resources are configured by three sets of tuning parameters that control the amount of shared memory (prefix SHM), number of semaphores (prefix SEM), and size of message queues and messages (prefix MSG).

The settings for these parameters are WebLogic Enterprise system dependent. Most UNIX systems, however, are shipped with default values that are too low for WebLogic Enterprise systems.

The following sections describe the IPC parameters and provide guidelines for configuring them. Because these parameters vary across different versions of UNIX, the following descriptions are generic. For the exact parameter names, default settings, settings used for the University Sample applications for each platform, and information about how to change the parameters, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
If you change a parameter, you need to rebuild the kernel and reboot the operating system using the standard administrative tools. Consult your operating system administrator or the system administrator’s guide for your platform for details.

If your WebLogic Enterprise client or server application is distributed, the minimum IPC resources must be available on every UNIX platform participating in the application.

**Semaphores**

Every process that participates in a WebLogic Enterprise system requires a semaphore. When the system boots, the number of semaphores configured in the operating system is checked, and the boot fails if the configured number is not high enough.

Semaphores on UNIX systems are grouped in semaphore sets. Each semaphore in a set can be accessed separately. Although WebLogic Enterprise does not perform operations on semaphore sets, it attempts to allocate as many semaphores per semaphore set as possible. WebLogic Enterprise also needs undo structures to function properly. The operating system uses undo structures to unlock semaphores held by a process that dies unexpectedly.

The following semaphore parameters may need to be adjusted:

**SEMMNS**

Maximum number of semaphores in the system. The minimum requirement for SEMMNS is:

```
MAXACCESSERS - MAXWSCLIENTS + 13
```

where MAXACCESSERS is the maximum number of WebLogic Enterprise processes on a particular machine (including servers and native clients), and MAXWSCLIENTS is the maximum number of WebLogic Enterprise remote clients. Both of these parameters are specified in the application’s UBBCONFIG file.

For more information about UBBCONFIG, see Creating a Configuration File in the WebLogic Enterprise online documentation, or the `ubbconfig(5)` reference page in the *BEA Tuxedo Reference Manual*.

**SEMMNI**

Maximum number of active semaphore sets. See SEMMSL.
Preparing to Install the WebLogic Enterprise Software

SEMMSL
Maximum number of semaphores per semaphore set. SEMNI and SEMMSL are commonly chosen so that their product equals SEMNS. The WebLogic Enterprise system does not perform semaphore operations on semaphore sets; however, it attempts to allocate as many semaphores per semaphore set as possible.

SEMMAP
Size of the control map used to manage semaphore sets. SEMMAP should be equal to SEMNI.

SEMNU
Number of undo structures in the system. Because an undo structure is needed for each process that can access the Bulletin Board, SEMNU must be at least as large as SEMNS.

SEMUME
Maximum number of undo entries per undo structure. The value 1 suffices.

Message Queues and Messages

WebLogic Enterprise client and server applications use UNIX messages and message queues for client/server communication. Examples of such messages are service requests, service replies, conversational messages, unsolicited notification messages, administrative messages, and transaction control messages.

Every Multiple Servers, Single Queue (MSSQ) set of servers, and every individual server has a message queue for receiving requests. Every client has its own queue for receiving replies. Servers that specify the REPLYQ parameter also get individual reply queues.

The adjustment of kernel message parameters is important to the proper tuning of the WebLogic Enterprise system. Inappropriate values can lead to an inability to boot, or to severe performance degradation.

There are various message queue parameters. They limit various characteristics of the queue space, including the total number of outstanding messages (MSGTQL), the total number of bytes that can be on one queue (MSGMNB), the size limit of an individual message (MSGMAX), the total number of message segments that can be outstanding at one time (MSGSEG), and the size of each segment (MSGSSZ).
Exceeding any of the parameter limits described previously results in what is known as a blocking condition. There is a special case for `MSGMAX`. Messages that exceed 75 percent of `MSGMNB`, or that are larger than `MSGMAX`, are placed in a UNIX file. A very small message with the filename in it is then sent to the recipient. Avoid this mode of operation, because it results in a severe reduction in performance.

An application deadlock can result if every process is blocked when it tries to send a message. For example, when client applications fill the message space with requests, and server applications are all blocked when they try to send replies, because no server application can read a message, there is a deadlock. Timeouts can sometimes break the deadlock, but no useful work will have been done.

Especially troublesome is a client application that sends its requests with the `TPNOREPLY` flag. This practice can fill either individual queues or the system message space, depending on the size of the messages. Such applications may have to implement their own flow control to limit the number of outstanding messages.

To summarize, if client applications or server applications are blocking on their send operations (that is, requesting services or sending replies), there is potential for trouble. It is usually no problem, though, for a single server request queue to always be full, as long as there is space in the system for more messages on other queues.

There are performance implications to queue blocking conditions, both on the sending side and the receiving side. The UNIX operating system, when waking up blocked processes, wakes up all the processes blocked on a particular event, even if only one can proceed. The other processes go back to sleep. This process scheduling overhead can be expensive.

For example, on an empty server request queue where there is more than one server application (that is, MSSQ), an arriving message wakes up all the idle, or blocked, server applications on that queue. In the case of a full server request queue, as each request is read by a server application, the system wakes up all the blocked clients. Depending on the size of the messages, zero or more clients are allowed to place their messages on the queue. The remainder of the clients have to go back to sleep. Because there may be hundreds of clients in the system, the mass wakeup of all of these clients every time a service request is processed can severely degrade performance.

A properly tuned system rarely fills its queues. Enough slack should be left in the queues to handle the natural variability of the message flow. No exact settings can be recommended. Tuning is very system dependent. The UNIX `ipcs(1)` command provides a snapshot of the queues so you can tell whether they are full. You can try the
TPNOBLOCK flag when sending requests. That way, clients can tell when queues are full, and they can slow down a bit. It might help to increase the scheduling priority of the servers whose request queues are full.

The following message parameters may need to be adjusted:

**MSGMNI**

Number of unique message queue identifiers. Each process participating in a WebLogic Enterprise client or server application on a particular machine typically needs at least one message queue. This number is reduced if MSSQ sets are used, where multiple server processes share a single queue. For transaction processing, count an additional queue per server group for TMS processes. Thus, the minimum requirement for MSGMNI can be determined by this formula:

\[
\text{MSGMNI} = \text{MAXACCESSERS} + 7 + (\text{number of servers with } \text{REPLYQ}) + (\text{number of MSSQ sets}) - (\text{number of servers in MSSQ sets})
\]

**MSGMAX**

Maximum message size in bytes. MSGMAX must be large enough to handle any WebLogic Enterprise client or server application running on this machine.

**MSGMNB**

Maximum message queue length in bytes. This number must accommodate the total size of all messages that are on a queue and that have not been taken off by the associated process(es). The minimum value for MSGMNB is MSGMAX. Messages longer than 75 percent of MSGMNB are sent to a file instead of to a message queue. Avoid this situation because it severely degrades performance.

**MSGMAP**

Number of entries in the control map used to manage message segments. MSGMAP should be the same as the number of message segments (MSGSEG), which should be twice the size of MSGMNI.

**MSGSSZ**

Size of a message segment in bytes. A message can consist of several such segments. The value of MSGSSZ should be such that a multiple of MSGSSZ is equal to the size (including the WebLogic Enterprise system header) of the most commonly sent message. This practice avoids wasting space.
MSGSEG
Number of message segments in the system.

MSGTQL
Total number of outstanding messages that can be stored by the kernel. This is the maximum number of unread messages at any given time.

Shared Memory

In the WebLogic Enterprise environment, shared memory is used for the Bulletin Board and for the control table of the IIOP Listener. An application also may choose to use shared memory for its own purposes.

The following shared memory parameters may need to be adjusted:

SHMMAX
Maximum shared memory segment size in bytes. This number represents the largest shared memory segment that can be allocated. A process can, however, attach to more than one segment of size SHMMAX.

SHMSEG
Maximum number of shared memory segments per process. For a given configuration, the maximum amount of shared memory in bytes to which a process can attach is SHMMAX * SHMSEG. A value between 6 and 15 should be adequate.

SHMMNI
Maximum number of shared memory identifiers in the system. The WebLogic Enterprise system requires one identifier per Bulletin Board and an additional identifier if the IIOP Listener is running.

SHMMIN
Minimum shared memory segment size in bytes. This should always be set to 1.
Other Kernel Tuning Parameters

Experience with WebLogic Enterprise systems has shown that some other UNIX tuning parameters may need to be set to higher values. The settings are dependent on the application and do not apply to all applications.

ULIMIT
Maximum file size. ULIMIT needs to be large enough so that you can install the WebLogic Enterprise software and build servers. We recommend 4 MB.

NOFILES
Maximum number of open files per process. A WebLogic Enterprise server application requires a minimum of four file descriptors.

MAXUP
Maximum number of processes per non-super user. The WebLogic Enterprise system processes (servers and administrative processes) run with the UID specified in the application’s UBBCONFIG file. MAXUP needs to be large enough to allow all these processes to run.

NPROC
Maximum number of processes (system wide).

NREGION
Number of region table entries to allocate. Most processes have three regions: text, data, and stack. Additional regions are needed for each shared memory segment and shared library (text and data) attached. However, the region table entry for the text of a shared text program is shared by all processes executing that program. Each shared memory segment attached to one or more processes uses another region table entry.

NUMTIM
Maximum number of STREAMS modules that can be pushed by the Transport Layer Interface (TLI). A typical default value is 16. Set NUMTIM to at least 256.

NUMTRW
The number of TLI read/write structures to allocate in kernel data space. A typical default value is 16. Set NUMTRW to at least 256.
Calculating IPC Requirements

When the WebLogic Enterprise software has been installed and an application configuration file (UBBCONFIG file) is available, the tmloadcf command can be used to calculate the IPC resources needed to support the application. For more information, see the tmloadcf(1) reference page in the BEA Tuxedo Reference Manual. Also see “Verifying IPC Requirements” on page 4-17.
Preparing to Install the WebLogic Enterprise Software
Part II  T-Engine Installation

Chapter 2.  WebLogic Enterprise T-Engine Installation on Windows Systems
Chapter 3.  WebLogic Enterprise T-Engine Installation on UNIX Systems
Chapter 4.  WebLogic Enterprise T-Engine Postinstallation Considerations
Chapter 5.  BEA Administration Console Startup
Chapter 6.  WebLogic Enterprise T-Engine Platform Data Sheets
CHAPTER 2

2 WebLogic Enterprise T-Engine Installation on Windows Systems

This chapter explains how to install the WebLogic Enterprise 5.1 T-Engine software on Microsoft Windows 2000, NT 4.0, 98, and 95 systems.

This topic includes the following sections:

- Platforms Supported
- If You Are Upgrading from a Previous Release
- Installing the WebLogic Enterprise Software on Microsoft Windows 2000 or NT 4.0 Systems
- Installing the WebLogic Enterprise Software on Microsoft Windows 98 and 95 Systems
- Removing (Uninstalling) the WebLogic Enterprise Software from Your System

Note: Part of the installation procedure described in this chapter includes running a sample application to verify that the installation is successful. If you are installing either the CORBA Java or J2EE software, and you want to run this sample application, you need to have the Java 2 SDK version 1.2.2 software installed on your system.
Platforms Supported

The Microsoft Windows platforms listed in Table 2-1 are supported.

Table 2-1  Supported Microsoft Platforms

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Release/Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows 2000</td>
<td></td>
</tr>
<tr>
<td>Microsoft Windows NT</td>
<td>4.0 Service Pack 5 (SP5) on Intel</td>
</tr>
<tr>
<td>Microsoft Windows 98</td>
<td></td>
</tr>
<tr>
<td>Microsoft Windows 95</td>
<td>Service Pack 1</td>
</tr>
</tbody>
</table>

You can install all or selected WebLogic Enterprise server, client, and administration components on a Microsoft Windows 2000 or NT 4.0 SP5 (Intel) operating system. You can install only the WebLogic Enterprise client components on the Microsoft Windows 98 and 98 operating systems. Windows 98 and 95 systems cannot be used as WebLogic Enterprise server systems.

For the hardware and software requirements for these operating systems, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”

If You Are Upgrading from a Previous Release

If you are installing the WebLogic Enterprise 5.1 software on a Windows 2000, NT, Windows 98, or Windows 95 system that contains a previous version of WebLogic Enterprise, M3, or BEA Tuxedo software, there are important upgrade considerations.

BEA recommends that you use the Windows Add/Remove (uninstall) program to remove a previous version of WebLogic Enterprise, M3, or BEA Tuxedo software on the target system, before you install WebLogic Enterprise 5.1. During the WebLogic
If You Are Upgrading from a Previous Release

Enterprise 5.1 installation, the procedure will try to detect earlier versions of WebLogic Enterprise, M3, or BEA Tuxedo; if present, the procedure will recommend that you cancel the installation, exit Setup, and use the Windows Add/Remove program.

**Note:** BEA strongly recommends that you not run more than one version of WebLogic Enterprise, M3, or BEA Tuxedo software on the same system. Unexpected problems can occur.

Starting with a Clean System

To ensure that you are starting with a clean system, before you install WebLogic Enterprise 5.1, follow these steps:

1. Use the `tmshutdown` command to stop any running WebLogic Enterprise, M3, or BEA Tuxedo applications. This command is described in the Administration topics in the WebLogic Enterprise online documentation.

2. If necessary, stop the TListen and Tuxedo IPC Helper services. From the Start menu, click Start—>Settings—>Control Panel—>Services. A screen similar to the following is displayed.
3. Scroll to the entry for the TListen service, select it, and then click the Stop button. The Status value should change from Started to a blank entry. Then scroll to the Tuxedo IPC Helper service, select it, and click the Stop button. In some cases, you may see an error; however, the service’s Status value should change from Started to a blank entry. Click the Close button.

4. Move to a temporary location any files that you or your coworkers added to the %TUXDIR% directory, where TUXDIR is the directory in which the prior WebLogic Enterprise, M3, or BEA Tuxedo software resides. This step is necessary because the Windows Add/Remove uninstall program only knows about the original set of files that were installed by BEA. If additional files are present, older directories may continue to exist after you run the uninstall program.

5. Back up any existing WebLogic Enterprise, M3, or BEA Tuxedo files that you customized for your environment. For example, you should back up the Resource Manager (RM) file in %TUXDIR%\udataobj\Rm. The Rm file contains database vendor-specific settings that are used by commands such as buildtms and buildXAJs. You may also need to back up the BEA Administration Console webgui.ini initialization file to a temporary location. This file is located in %TUXDIR%\udataobj\webgui, where TUXDIR is the directory in which you installed the prior version of WebLogic Enterprise, M3, or BEA Tuxedo.
6. Run the Windows Add/Remove program to remove the prior WebLogic Enterprise, M3, or BEA Tuxedo software version. From the Start menu, click Start—>Settings—>Control Panel—>Add/Remove Programs. A screen similar to the following is displayed:

![Add/Remove Programs Properties](image)

7. Scroll to the entry for the prior WebLogic Enterprise, M3, or BEA Tuxedo software, select it, and click the Add/Remove button.

**Note:** If you have both the C++ and Java components of WebLogic Enterprise 4.2 installed on your machine, you must remove the Java software *before* removing the C++ software.
8. In response to the prompt, confirm that you want to uninstall the software. After the program finishes, it displays a screen similar to the following:

![Remove Programs From Your Computer](image)

9. If the uninstall program was not able to remove all directories (usually because the files were added after the original installation), you can click the Details button to find out which directories remain on your system. If the files in the directories contain changes that you made, such as a modified sample file, move it to a temporary location.

10. If the prior version of WebLogic Enterprise was 4.2 or 4.1, you may need to uninstall the WebLogic Enterprise Java and WebLogic Enterprise C++ software as separate steps. **Remember:** if you have both the C++ and Java components of WebLogic Enterprise 4.2 installed on your machine, you must remove the Java software *before* removing the C++ software, as shown in Step 7.

11. Reboot your system after the uninstall completes.
12. Install the WebLogic Enterprise 5.1 software, as described in this chapter. When the WebLogic Enterprise 5.1 software installation finishes, compare the files from a previous release that you moved to a temporary location (such as your RM file described in a previous step) with the installed version. If appropriate, customize the installed file so that it contains the data that is appropriate for your environment.

Installing the WebLogic Enterprise Software on Microsoft Windows 2000 or NT 4.0 Systems

This section describes how to install the WebLogic Enterprise software on Microsoft Windows 2000 or NT 4.0 systems.

Preinstallation Considerations

This section describes some important tasks that you should perform before starting the WebLogic Enterprise installation.

Backing Up Files

If you are installing WebLogic Enterprise software on a system that already has M3 or WebLogic Enterprise software installed, there are some files that you may want to back up prior to the installation, and then restore them after the installation is complete. This is because some files that you may have modified for your M3 or WebLogic Enterprise software are overwritten when the WebLogic Enterprise software is installed.

To avoid having to modify these files again, proceed as follows:
1. If you are installing one or more of the WebLogic Enterprise server software components, back up the RM file to a temporary location. This file is located in the %TUXDIR%\udataobj or $TUXDIR/udataobj directory, where TUXDIR is the directory in which you installed the M3 or WebLogic Enterprise software.

2. If you are installing the BEA Administration Console, back up the webgui.ini file to a temporary location. This file is located in the %TUXDIR%\udataobj\webgui or $TUXDIR/udataobj/webgui directory.

3. After the installation is complete, restore these files to their original locations.

**Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services**

Before beginning the installation, make sure no BEA Tuxedo or WebLogic Enterprise client or server applications are running. For information about the tmshutdown command, see Starting and Shutting Down Applications in the Administration section of the WebLogic Enterprise online documentation.

**Checking That Your Account Has Administrator Privilege**

You need administrator privileges to perform the installation. If you attempt to install the WebLogic Enterprise software without administrator privileges, the following error message will be displayed:

Cannot Install Tuxedo IPC Helper Service.

**Microsoft Windows 2000 or NT 4.0 Installation Procedure**

It will take approximately 10 minutes to install the software.

**Warning:** If you are re-installing the WebLogic Enterprise 5.1 software on your system, and you also already installed the optional WebLogic Enterprise 5.1 Encryption Package software (56-bit or 128-bit) on your system, you must:

a. First, uninstall the WebLogic Enterprise Encryption Package software. (See the section “Removing (Uninstalling) the WebLogic Enterprise Encryption Package from Your System” on page 12-14.)
b. Next, uninstall the WebLogic Enterprise 5.1 software. (See the section “Removing (Uninstalling) the WebLogic Enterprise Software from Your System” on page 2-43.)

c. Re-install the WebLogic Enterprise 5.1 software, as explained in this section.

To install the WebLogic Enterprise software on a Microsoft Windows 2000 or NT 4.0 operating system, complete the following steps:

1. Insert the BEA WebLogic Enterprise software CD in the CD player. The Windows autorun feature is used to automatically begin the installation. (To bypass the autorun feature, hold down the Shift key while inserting the CD in the CD player. You can then run the setup.exe file in the inwnt40 directory on the CD.)

2. The Setup screen is displayed, followed by the Welcome screen.

3. Click Next. The Software License Agreement screen is displayed.
4. To accept the license agreement, click Yes. The User Information screen is displayed.
5. Enter your name and the name of your company and click Next. The Registration Confirmation screen is displayed.
6. If the registration information is correct, click Yes; otherwise, click No and correct the information. The Choose Destination Location screen is displayed.

The default destination folder is `c:\wledir`. In the previous sample screen, the user selected `d:\wledir` (or this was the location of previously installed software).

7. Click Next to accept the location, or click Browse to select a different location. If you enter a path that does not exist, the installation procedure prompts for a confirmation that you want the directory created.

After you complete the directory path screen, the BEA WebLogic Enterprise Software Component Selection screen is displayed.
8. By default, all WebLogic Enterprise components are selected for installation. The WebLogic Enterprise 5.1 installation procedure lets you select or deselect the components that you want to install. You can also select or deselect specific subcomponents within the Servers or Clients categories. The BEA Administration Console component does not have any subcomponents.

You must select at least one component for installation. Selecting a main component category causes all of its subcomponents to be selected. Deselecting a component causes all of its subcomponents to be deselected. Deselecting all subcomponents causes their parent component to be deselected.

The main WebLogic Enterprise categories for the installation are:

- Servers
- Clients
- BEA Administration Console
To deselect or select particular servers for installation, **highlight the Servers category** and then click the Change... button. On the Select Sub-components screen, deselect the server types you want to omit. Within the Servers category, the options are:

- BEA Tuxedo Server
- BEA CORBA C++ Server
- BEA CORBA Java Server
- BEA J2EE Server

**Notes:** The BEA Tuxedo Server is always installed in a Server installation, even if you deselect it on this screen.

When you install an individual type of server software, its corresponding client software is also installed. For example, if you install the J2EE Servers software, the RMI/EJB client software is also installed automatically. Therefore, if you have installed a type of server software, you do not need to perform a separate installation for the client software for that server.

On the Select Sub-components screen, click the Continue button when you have made your choices. For example:
After you click the Continue button, you are returned to the BEA WebLogic Enterprise Software Component Selection screen.

To deselect or select particular clients for installation, **highlight the Clients category** and then click the Change... button. On the Select Sub-components screen, deselect the client types you want to omit. The Client options are:

- BEA Tuxedo Client
- BEA CORBA C++ Client
- BEA RMI/EJB Client
- BEA ActiveX Client
- BEA CORBA Java Client

On the Select Sub-components screen for the clients, click the Continue button when you have made your choices. For example:
You are returned to the BEA WebLogic Enterprise Software Component Selection screen.

The Administration category consists of the Administration Console and does not have any subcomponents. For information about how to start this Console after it is installed, refer to Chapter 5, “BEA Administration Console Startup.” For information about how to use this Console, refer to the online help that is accessible through the Console’s Help button.

When you have made your selections, click Next. If you indicated that your installation will include the BEA Administration Console, the Choose Destination Location screen for the Administration Console HTML files is displayed.
9. For information about the path on this screen, see the section “For All Server Platforms Supporting the BEA Administration Console” on page 1-15. To specify a nondefault directory for the BEA Administration Console HTML files, click Browse, specify the nondefault directory, and click Next. Otherwise, click Next. Another Choose Destination Location screen is displayed.
10. For information about the path on this screen, see the section “For All Server Platforms Supporting the BEA Administration Console” on page 1-15. To specify a nondefault directory for the BEA Administration Console GUI Java applets, click Browse, specify the nondefault directory, and click Next. Otherwise, click Next. Another Choose Destination Location screen is displayed.
11. For information about the path on this screen, see the section “For All Server Platforms Supporting the BEA Administration Console” on page 1-15. To specify a nondefault directory for the BEA Administration Console GUI CGI programs, click Browse, specify the nondefault directory, and click Next. Otherwise, click Next. The Enter Information screen is displayed.
12. For information about the path on this screen, see the section “For All Server Platforms Supporting the BEA Administration Console” on page 1-15. To specify a nondefault Web Server client prefix for the GUI CGI directory, click Browse, specify the nondefault prefix, and click Next. Otherwise, click Next. The Select Program Folder screen is displayed.
13. To specify a nondefault program folder name, enter the folder name or select a folder name from the Existing Folders, and click Next.

14. If the installation program detects a prior version of M3, WebLogic Enterprise, or BEA Tuxedo (or an earlier installation of WebLogic Enterprise 5.1) on the target system, the installation program recommends that you cancel the installation, exit Setup, and run the Windows Add/Remove program to remove it. For example:
BEA recommends that you not run multiple versions of WebLogic Enterprise, M3, or BEA Tuxedo on the same machine. If you see this prompt, click the OK button, then click the Cancel button in the next screen:

If you had to click the Cancel button because the installation program detected existing M3, WebLogic Enterprise, or BEA Tuxedo software on the target system, the installation program displays the following screen. Click the Exit Setup button, and then use the Windows Add/Remove program to remove the prior installation, as described in the section “Starting with a Clean System” on page 2-3.

15. If you did not have to cancel the installation, the installation continues and you are prompted about having the installation program back up the target system’s performance data:
16. To back up your performance data files before you install the WebLogic Enterprise performance monitor, click Yes. The BEA WebLogic Enterprise TListen Password screen is displayed:

```
BEA WebLogic Enterprise TLlisten Password
Password: ********
Confirm: ********
```

17. Enter the *tl*isten password in the Password field and again in the Confirm field and click Ok. If Cancel is clicked, the *tl*isten password installation is deferred. For information about the *tl*isten password and instructions for setting it, see the section “Selecting an Administrative Password” on page 1-17. The Install license screen is displayed.

```
Install license.
Would you like to install your license now?
```

18. To install the WebLogic Enterprise software license now, click Yes; otherwise, click No to install the license later. If you click Yes, the Insert License Disk screen is displayed.
19. Your product license is on a 3.5-inch diskette that is included in the WebLogic Enterprise product box. To install the license, insert the license diskette in the disk drive on your machine and, if your diskette drive is drive A, click OK; otherwise, enter the correct drive and click OK. An Information screen is displayed informing you that the WebLogic Enterprise license file installed successfully.

**Note:** If you decide that you do not want to install the license now, but you want to complete the installation procedure and install the license later, do not click Cancel. Clicking Cancel terminates the installation. Instead, remove the license diskette from the disk drive and click OK. A screen is displayed that states that the lic.txt could not be found and you can elect to complete the installation without installing the license.

20. Click OK. The installation procedure displays the following screen.
Click the Finish button to complete the installation.

21. Reboot your system.

**Setting Microsoft Windows 2000 or NT 4.0 Environment Variables**

Before you use the WebLogic Enterprise software, you may need to define the `JDK_HOME` environment variable.

The `JAVA_HOME` variable is needed to start Java server applications and to build and run Java sample applications.

To set this variable, complete the following procedure:
1. Click the Windows Start button, and click Settings—>Control Panel—>System—>Environment.

The System Properties screen is displayed.

2. Enter the `JAVA_HOME` environment variable and set its value to the directory containing the JDK, as shown in the sample screen.

   **Note:** JavaServer will not start on Microsoft Windows NT or Windows 2000 if JDK bin is in the path after a network drive. Make sure the JDK bin directories (that is, jre/bin and jre/bin/classic) are set in the PATH before
any network driver path elements via the Control Panel before booting the
JavaServer.

3. Click Apply and OK to close the System Properties window.

Installing the Product License After You Install the
WebLogic Enterprise Software

If you elected not to install your software license when you installed the WebLogic
Enterprise software, you can install the license using the BEA License Utility.

Note: Your product license is on a 3.5-inch disk that is included in the software box.

To install the license, complete the following steps:

1. Insert the license disk into the disk drive on your machine.

2. Use the taskbar to click Start—>Programs—>BEA WebLogic Enterprise 5.1
   —>BEA License Utility 5.1.

   The BEA License Utility screen is displayed.

   ![BEA License Utility]

   Please enter source file containing license text and
target license file for your installed product.

   Source File:
   A:\lic.txt

   Target File:
   \[your_BEAM_softwar\directory\data\lic

   [Browse] [OK] [Cancel]
3. If the disk drive on your machine is drive A, click OK; otherwise, enter the correct drive and click OK. The license is installed and the License File updated message is displayed.

Running Simpapp to Verify the WebLogic Enterprise CORBA ++ Software Installation on Microsoft Windows 2000 or NT 4.0

To verify that you have successfully installed the WebLogic Enterprise CORBA C++ client and server software, execute the simpapp application. This "simple application" is a WebLogic Enterprise client/server application that converts text strings to uppercase and lowercase letters, and can verify whether your installation is successful.

**Note:** This section assumes you installed all WebLogic Enterprise server components, or one of the CORBA server components. If you installed only the J2EE server component, see the next section, “Running a Basic EJB Sample to Verify the WebLogic Enterprise J2EE Software Installation” on page 2-35, for information about running an EJB sample to verify the installation.

Before attempting to run simpapp, refer to the section “Software Requirements” on page 6-25 to ensure that the software requirements are satisfied. For example, the path to the Microsoft Visual C++ 6.0 environment must be known on this system, because nmake is used.

To run simpapp, open an MS-DOS window and complete the following steps:
1. Make sure that the directory in which you installed the WebLogic Enterprise software is set in the environment variable TUXDIR. For example, if you installed the software in the default directory, complete the following steps to set the TUXDIR environment variable to C:\WLEDIR:

a. On the Windows Start menu, click Settings—>Control Panel. The Control Panel is displayed.

b. Click the BEA Administration icon. The BEA Administration screen is displayed.
c. If the BEA Administration screen is not displaying the Environment page as shown in the above screen, click the Environment tab. The Environment page is displayed.

d. Click on the TUXDIR variable, enter C:\WLEDIR in the value field, and click OK.

2. Using Windows Explorer, create a directory under WLEDIR and copy the contents of the simpapp directory to it. If you installed the WebLogic Enterprise software in the default directory, the simpapp directory is located at C:\WLEDIR\Samples\Corba\Simpapp.

3. Using an MS-DOS window:
   a. Change (cd) to the copy directory.
   b. Check the permissions on all the files in the copy directory and, if necessary, change the permissions to allow full access. To set permissions to full access, enter attrib -R /S *. *

2-30   Installation Guide
4. To run simpapp automatically, enter runme. The simpapp application runs and prints the following messages:

   Testing simpapp
   cleaned up
   prepared
   built
   loaded ubb
   booted
   ran
   shutdown
   saved results
   PASSED

5. To run the sample manually to observe the simpapp processes starting and stopping, complete the following steps:

   a. Enter results\setenv.

   b. Enter tmboot -y. The application starts several processes.

   c. Enter simple_client. The prompt String? is displayed.

   d. Enter a word in lowercase letters. The application converts the word to uppercase and then to lowercase letters.

   e. Enter tmshutdown -y. The application shuts down the processes.

6. To restore the directory to its original state, enter the following:

   a. results\setenv

   b. nmake -f makefile.nt clean
Running Simpapp to Verify the WebLogic Enterprise CORBA Java Software Installation on Microsoft Windows 2000 or NT 4.0

To verify that you have successfully installed the WebLogic Enterprise CORBA Java client and server software, execute the simpapp.java application. This “simple application” is a WebLogic Enterprise client/server application that converts text strings to uppercase and lowercase letters, and can verify whether your installation is successful.

Notes: This section assumes you installed all WebLogic Enterprise server components, or the CORBA Java server components. If you installed only the J2EE server component, see the next section, “Running a Basic EJB Sample to Verify the WebLogic Enterprise J2EE Software Installation” on page 2-35, for information about running an EJB sample to verify the installation.

If you have installed the CORBA Java server components, the CORBA C++ components are also automatically installed. Therefore, if running simpapp.java as described in this section is successful, it also verifies that the CORBA C++ components were installed correctly.

Before running the sample application, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets” for important information about prerequisite software. For example, on NT systems, you must first download and install the Java 2 SDK version 1.2.2.

Before attempting to run simpapp.java, refer to the section “Software Requirements” on page 6-25 to ensure that the software requirements are satisfied. For example, the path to the Microsoft Visual C++ 6.0 environment must be known on this system, because nmake is used.

To run simpapp, open an MS-DOS window and complete the following steps:

1. Make sure that the directory in which you installed the WebLogic Enterprise software is set in the environment variable TUXDIR. For example, if you installed the software in the default directory, complete the following steps to set the TUXDIR environment variable to C:\WLEDIR:
   a. On the Windows Start menu, click Settings—>Control Panel. The Control Panel is displayed.
Installing the WebLogic Enterprise Software on Microsoft Windows 2000 or NT 4.0

b. Click the BEA Administration icon. The BEA Administration screen is displayed.
WebLogic Enterprise T-Engine Installation on Windows Systems

2. Using Windows Explorer, create a directory under \WLEDIR and copy the contents of the simpapp directory to it. If you installed the WebLogic Enterprise software in the default directory, the simpapp_java directory is located at C:\WLEDIR\Samples\Corba\Simpapp_java.

3. Using an MS-DOS window:
   a. Change (cd) to the copy directory.
   b. Check the permissions on all the files in the copy directory and, if necessary, change the permissions to allow full access. To set permissions to full access, enter attrib -R /S *. *

   c. If the BEA Administration screen is not displaying the Environment page as shown in the above screen, click the Environment tab. The Environment page is displayed.
   d. Click on the TUXDIR variable, enter C:\WLEDIR in the value field, and click OK.
4. To run simpapp.java automatically, enter runme. The simpapp.java application runs and prints the following messages:

   Testing simpapp
   cleaned up
   prepared
   built
   loaded ubb
   booted
   ran
   shutdown
   saved results
   PASSED

5. To run the sample manually to observe the simpapp.java processes starting and stopping, complete the following steps:

   a. Enter results\setenv.
   b. Enter tmboot -y. The application starts several processes.
   c. Enter simple_client. The prompt String? is displayed.
   d. Enter a word in lowercase letters. The application converts the word to uppercase and then to lowercase letters.
   e. Enter tmshutdown -y. The application shuts down the processes.

6. To restore the directory to its original state, enter the following:

   a. results\setenv
   b. nmake -f makefile.nt clean

Running a Basic EJB Sample to Verify the WebLogic Enterprise J2EE Software Installation

If you installed only the WebLogic Enterprise J2EE server component, you can run the stateless session EJB sample application provided by the WebLogic Enterprise software to verify the installation.
Before running the sample application, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets” for important information about prerequisite software.

This sample demonstrates the usage of a stateless session EJB using a simple stock trader application. This sample demonstrates how the client must maintain any persistent state—such as the change in the cash account—across repeated calls to the session EJB. All the logic for the balance is encapsulated in the client, where all the persistence is provided by the container and the logic is maintained in the EJB.

The EJB in this sample provides basic trading methods, such as buying and selling stocks. Because there are no persistent stores involved in this sample, all the stock data are set in the deployment descriptor of the EJB as environment properties. The container supplies the data to the EJB through a JNDI lookup operation.

To build the stateless session EJB sample application, use the following steps:

1. Make sure that the directory in which you installed WebLogic Enterprise is set in the environment variable TUXDIR. Make sure to set the JAVA_HOME environment variable.

2. Make a copy of the %TUXDIR%\samples\j2ee\ejb directory into a work directory.

3. Change directory to the work directory.

4. Change the permissions on all the files to give them write-access. For example:

   ```prompt>attrib /S -r *```

5. Run the JavaServer version of the sample automatically by entering the runme command:

   ```prompt>runme basic statelessSession```

6. A number of messages are displayed, along with information about whether the build procedure was successful. The sample is built, the servers are booted, and the client is run once.

After you have executed the runme command, you can run the samples manually if you like.

To run the samples manually:

1. Change the current directory to the basic\statelessSession directory.
2. Make sure that your environment is set correctly by entering the following command:
   prompt>setenv

3. Boot the server, run the client, and shut down the server by entering the following commands:
   prompt>tmboot -y
   prompt>run_client.cmd
   prompt>tmshutdown -y

To restore the sample application directory to its original state:

1. Change to the work directory.

2. Enter the following command, where TUXDIR is the directory in which you installed the WebLogic Enterprise software:
   prompt>%TUXDIR%\samples\j2ee\ejb\clean.cmd

For more information about the stateless session EJB sample application, see the Guide to the EJB Sample Applications in the WebLogic Enterprise online documentation.

Installing the WebLogic Enterprise Software on Microsoft Windows 98 and 95 Systems

Notes: Before beginning the installation, ensure that no BEA Tuxedo or WebLogic Enterprise applications are running.

It takes approximately 10 minutes to install the software.

Warning: If you are re-installing the WebLogic Enterprise 5.1 software on your system, and you already installed the optional WebLogic Enterprise 5.1 Encryption Package software (56-bit or 128-bit) on your system, you must:
a. First, uninstall the WebLogic Enterprise Encryption Package software. (See the section “Removing (Uninstalling) the WebLogic Enterprise Encryption Package from Your System” on page 12-14.)

b. Next, uninstall the WebLogic Enterprise 5.1 software. (See the section “Removing (Uninstalling) the WebLogic Enterprise Software from Your System” on page 2-43.)

c. Re-install the WebLogic Enterprise 5.1 software, as explained in this section.

To install the BEA WebLogic Enterprise software on a Microsoft Windows 98 or 95 operating system, follow the steps listed below.

**Note:** In this section, installation screens that are identical to the screens shown in the Windows 2000 and NT 4.0 section are not repeated.

1. Insert the BEA WebLogic Enterprise software CD in the CD player. The Windows autorun feature is used to automatically begin the installation. (To bypass the autorun feature, hold down the Shift key while inserting the CD in the CD player. You can then run the setup.exe file in the inwnt40 directory on the CD.)

2. The Setup screen is displayed, followed by the Welcome screen.

3. Click Next. The Software License Agreement screen is displayed.

4. To accept the license agreement, click Yes. The User Information screen is displayed.

5. Enter your name and the name of your company and click Next. The Registration Confirmation screen is displayed.

6. If the registration information is correct, click Yes; otherwise, click No and correct the information. The Choose Destination Location screen is displayed.

7. The default destination folder is c:\wledir. Click Next to accept this location, or click Browse to select a different location. If you enter a path that does not exist, the installation procedure prompts for a confirmation that you want the directory created.

8. After you complete the directory path screen, the BEA WebLogic Enterprise Software Component Selection screen is displayed. For an installation on a Windows 98 or Windows 95 system, you can only install components from the client list.
To deselect or select particular clients for installation, click the Change... button. On the Select Sub-components screen, deselect the client types you want to omit. The Client options are:

- BEA Tuxedo Client
- BEA CORBA C++ Client
- BEA RMI/EJB Client
- BEA ActiveX Client
- BEA CORBA Java Client

On the Select Sub-components screen for the clients, click the Continue button when you have made your choices. Then click the Next button on the Software Components Selection screen.
9. The Select Program Folder screen is displayed. To specify a nondefault program folder name, enter the folder name or select a folder name from the Existing Folders, and click Next.

10. If a prior version of WebLogic Enterprise exists on your system, the installation procedure prompts you to remove it.

11. The WebLogic Enterprise software is installed. When the software installation completes, the Update AUTOEXEC.BAT screen is displayed.

12. Click Yes to update the AUTOEXEC.BAT file. The Set APPDIR environment variable screen is displayed.

13. Click Yes. The WebLogic Enterprise Application Directory screen is displayed.
14. To accept the default directory, click Next. To specify a nondefault directory, click Browse, enter the desired directory, and click Next. If the BEA Tuxedo Client was selected, the Set WSENVFILE environment variable screen is displayed.

15. To set the WSENVFILE environment variable, click Yes. The WebLogic Enterprise /WS Environment File screen is displayed.
16. To accept the default directory, click Next. To specify a nondefault directory, click Browse, enter the desired directory, and click Next. The Setup is verifying installation screen is displayed.

17. After the installation program verifies the installation, the installation is completed. If you had the installation program update your AUTOEXEC.BAT file, the following screen is displayed.
18. Click the Finish button and **reboot your system**.

**Removing (Uninstalling) the WebLogic Enterprise Software from Your System**

This section explains how to remove the WebLogic Enterprise software from your system.

**Warning**: If you also installed the optional WebLogic Enterprise Encryption Package software (56-bit or 128-bit) on your system, you must:
First uninstall the WebLogic Enterprise Encryption Package software. (See the section “Removing (Uninstalling) the WebLogic Enterprise Encryption Package from Your System” on page 12-14.)

Next, uninstall the WebLogic Enterprise software, as explained in this section.

To remove the WebLogic Enterprise software from your system:

1. Log on to the system. If you are using a Microsoft Windows 2000 or NT 4.0 system, log on as the administrator or as a member of the Administrator group.

2. Make sure that no BEA Tuxedo or WebLogic Enterprise client or server applications are running. Use `tmshutdown` to shut down all WebLogic Enterprise applications.

3. On the Microsoft Windows taskbar, click Start—>Programs—>BEA WebLogic Enterprise 5.1—>UnInstall BEA WebLogic Enterprise 5.1. The Confirm File Deletion screen is displayed:

   ![Confirm File Deletion](image)

   Are you sure you want to completely remove 'BEA WebLogic Enterprise 5.1' and all of its components?

   [Yes] [No]

4. Click Yes to confirm the removal and uninstall the WebLogic Enterprise software. The Remove Programs From Your Computer screen is displayed.
The WebLogic Enterprise product is removed from your system and from the Windows Registry. **Reboot your system.**
WebLogic Enterprise T-Engine Installation on Windows Systems
CHAPTER

3  WebLogic Enterprise
T-Engine Installation
on UNIX Systems

This topic includes the following sections:

- Platforms Supported
- Installing the WebLogic Enterprise Software on UNIX Systems
- Removing (Uninstalling) the WebLogic Enterprise Software from Your System

Note: Part of the installation procedure described in this chapter includes running a sample application to verify that the installation is successful. If you are installing either the CORBA Java or J2EE software, and you want to run this sample application, you need to have the Java 2 SDK version 1.2.2 software installed on your system.
Platforms Supported

The platforms listed in Table 3-1 are supported.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Operating System</th>
<th>Release/Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compaq</td>
<td>Tru64 UNIX</td>
<td>4.0f on Alpha systems</td>
</tr>
<tr>
<td>HP</td>
<td>HP-UX</td>
<td>11.00 32-bit plus patches B.11.00.B0315</td>
</tr>
<tr>
<td>IBM</td>
<td>AIX</td>
<td>4.3.3</td>
</tr>
<tr>
<td>SCO</td>
<td>UnixWare</td>
<td>7.1.1 (C++ only)</td>
</tr>
<tr>
<td>Sun Microsystems</td>
<td>Solaris</td>
<td>2.6 and 7 (SPARC)</td>
</tr>
</tbody>
</table>

Notes: The WebLogic Enterprise 5.1 software for Solaris is available in two varieties, depending upon whether you are building WebLogic Enterprise applications with either the Standard Mode or Compatibility Mode of the Sun Workshop Compiler C++ 5.0 software. For important details about how to choose the version of WebLogic Enterprise 5.1 for Solaris you need, see step 6 in the section “UNIX Installation Procedure” on page 3-4.

If you install the WebLogic Enterprise software on a Solaris system, there is a way to tell which version of WebLogic Enterprise (Standard Mode or Compatibility Mode) was installed. The /udataobj directory under the installed WebLogic Enterprise directory will contain either a CompatibilityMode.txt file or a StandardMode.txt file. The file has one line that states: You have installed components for Sun Solaris v<OS-version> <mode-type>.

For the hardware and software requirements for these operating systems, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
Installing the WebLogic Enterprise Software on UNIX Systems

This section describes how to install the WebLogic Enterprise software on UNIX systems.

Preinstallation Considerations

This section describes some important tasks that you should perform before starting the WebLogic Enterprise installation.

Backing Up Files

If you are installing WLE software on a system that already has M3 or WebLogic Enterprise software installed, there are some files that you may want to back up prior to the installation, and then restore them after the installation is complete. This is because some files that you may have modified for your M3 or WebLogic Enterprise software are overwritten when the WebLogic Enterprise software is installed.

To avoid having to modify these files again, proceed as follows:

1. If you are installing one or more of the WLE server software components, back up the RM file to a temporary location. This file is located in the TUXDIR/udataobj directory where $TUXDIR is the directory in which the prior WebLogic Enterprise or M3 software exists.

2. If you are installing the BEA Administration Console, back up the webgui.ini file to a temporary location. This file is located in the M3DIR/udataobj/webgui or WLEDIR/udataobj/webgui directory.

3. After the installation is complete, restore these files to their original locations.
WebLogic Enterprise T-Engine Installation on UNIX Systems

Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services

Before beginning the installation, make sure no BEA Tuxedo or WebLogic Enterprise client or server applications are running. For information about the \texttt{tmshutdown} command, see \textit{Starting and Shutting Down Applications} in the \textit{Administration} section of the WebLogic Enterprise online documentation.

Checking That Your Account Has the Required Privileges

On most systems, you need superuser privileges to mount the software CD. The account that you log on to to perform the installation must have administrative privileges.

UNIX Installation Procedure

It takes approximately 10 minutes to install the software.

\textbf{Warning:} If you are re-installing the WebLogic Enterprise 5.1 software on your system, and you also already installed the optional WebLogic Enterprise Encryption Package software (56-bit or 128-bit) on your system, you must:

- First, uninstall the WebLogic Enterprise 5.1 software on your UNIX system, which also removes the WebLogic Enterprise Encryption Package software. (See the section “Removing (Uninstalling) the WebLogic Enterprise Software from Your System” on page 3-19.)
- Next, re-install the WebLogic Enterprise 5.1 software, as explained in this section.

To install the WebLogic Enterprise software on a UNIX operating system, complete the following steps:

1. Log on to the system with administrative privileges.
2. Insert the WebLogic Enterprise CD into the reader.
3. Mount the CD as a file system. For platform-specific instructions on how to do this, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.” On most systems, you need superuser privileges to perform the mount.
Note: If your system does not have a directly connected CD reader, you can mount the CD on a remote system, share (export) the CD file system, and then mount the remote file system. For detailed instructions for each platform, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.” Alternatively, you can mount the CD on a remote system, copy the contents of the CD directory for your platform to the system in which you plan to install the WebLogic Enterprise software, and continue with the remainder of the installation procedure.

4. Use the `cd` command to change your working directory to the root of the WebLogic Enterprise 5.1 software CD.

5. Run the `ls` command in the root directory to check the CD’s contents. If all the files are in lowercase characters, begin the installation by entering:
   ```bash
   sh install.sh
   ```
   If all the files are in uppercase characters, begin the installation by entering:
   ```bash
   sh INSTALL.SH
   ```

6. The installation procedure lists the available platform choices. Enter the number that corresponds to the installation’s target platform.

   **Note:** If you are installing the WebLogic Enterprise 5.1 software on a Solaris 2.6 system or a Solaris 7 system, the installation procedure displays a prompt asking you to choose the version of WebLogic Enterprise 5.1 that is compatible with the applications and software components you intend to use on the Solaris system. The prompt offers the following choices:

   - **Standard Mode**—this version of WebLogic Enterprise 5.1 has been built with the Sun Workshop 5.0 C++ compiler in Standard Mode. Choose this if your WebLogic Enterprise applications need to be compatible with other applications and components that have been built with the Sun Workshop 5.0 C++ compiler in Standard Mode.

   - **Compatibility Mode**—this version of WebLogic Enterprise 5.1 has been built with the Sun Workshop 5.0 C++ compiler in Compatibility Mode. Choose this if your WebLogic Enterprise applications need to be compatible with WebLogic Enterprise 4.2 or other applications and components that have been built with the Sun Workshop 4.2 C++ compiler, or with the 5.0 compiler in Compatibility Mode. (Refer to the Sun Workshop 5.0 C++ compiler documentation for details on the use and support of compatibility mode.)
For example, the prompts during the WebLogic Enterprise installation on Solaris are:

3) Sun Solaris v2.6 (Standard Mode)  
4) Sun Solaris v2.6 (Compatibility Mode)  

Or:

5) Sun Solaris 7 (Standard Mode)  
6) Sun Solaris 7 (Compatibility Mode)  

**Note:** If you install the WebLogic Enterprise software on a Solaris system, there is a way to tell which version of WebLogic Enterprise (Standard Mode or Compatibility Mode) was installed. The /udataobj directory under the installed WebLogic Enterprise directory will contain either a CompatibilityMode.txt file or a StandardMode.txt file. The file has one line that states: You have installed components for Sun Solaris v<OS-version> <mode-type>.  

7. The remaining prompts in this chapter show a sample WebLogic Enterprise 5.1 installation on a Solaris 2.6 system. For example, a confirmation prompt is displayed. (In this example, assume we indicated in the previous step that we want to install the Solaris 2.6 Compatibility Mode version.)

**You have chosen to install software for**  
BEA WebLogic Enterprise Release 5.1  
This directory contains the BEA WLE Installation Software for Sun Solaris v2.6 on Sun SPARC (Compatibility Mode).  
Is this correct? [y,n,q]:

8. Enter y to accept the selection; or enter n to reject the selection and return to the list of platforms; or enter q to quit the installation.  

9. An informational messages and the initial component selection menu are displayed:  
To terminate the installation at any time press the interrupt key, typically <del>, <break>, or <ctrl+c>.  
The following components are available:

<table>
<thead>
<tr>
<th>Number</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>servers</td>
<td>BEA WebLogic Enterprise Servers</td>
</tr>
<tr>
<td>2</td>
<td>clients</td>
<td>BEA WebLogic Enterprise Clients</td>
</tr>
</tbody>
</table>
3     admcon          BEA Administration Console
Select the one you wish to install [?,??,q]:

The server and client components have packages that you can select for a more specific installation. Enter the number 1 to display all the WebLogic Enterprise server packages; or enter 2 to display all the WebLogic Enterprise client packages; or enter 3 to display the Administration Console component; or enter a single question mark (?) to display a brief help message; or enter two question marks (??) to redisplay the menu; or enter q to quit the installation.

10. **If you entered the number 1** in the component selection menu to view the list of WebLogic Enterprise server packages, the following menu is displayed:

The following packages are available:

1     wletsrv    BEA WebLogic Enterprise Tuxedo Server
2     wlecsrv    BEA WebLogic Enterprise CORBA C++ Server
3     wlejsrv    BEA WebLogic Enterprise CORBA Java Server
4     wlej2ee    BEA WebLogic Enterprise J2EE Server

Select the package(s) you wish to install (or ‘all’ to install all packages) (default: all) [?,??,q]:

Select the server package you want to install. (Although the prompt implies that you could enter different combinations of server packages, you can only enter a single type of server during each installation procedure, or you can enter all, the default option, to install all packages at once.)

**Note:** When you select a server package, that server package and its corresponding client packages are installed. The Administration Console component is not installed. If you later want to install the Administration Console on your system, you can run the installation procedure again to select and install the Administration Console.

**Note:** The BEA Tuxedo server software is always installed as a base component for any of the other WebLogic Enterprise servers. Also, the CORBA Java and J2EE server components are always installed together, even if you only select one of those items.

If instead you entered 2 in the component selection menu to view the list of WebLogic Enterprise client packages, the following menu is displayed:

The following packages are available:

1     wletcli   BEA Tuxedo Client
Select the package(s) you wish to install (or 'all' to install all packages) (default: all) [?,??,q]:

Select the client package you want to install. Or enter all, the default option, to install all the client packages.

If instead you entered 3 in the initial component selection menu, the following menu is displayed:

The following packages are available:

1     admcon     BEA Administration Console

Select the package(s) you wish to install (or 'all' to install all packages) (default: all) [?,??,q]:

Enter the number 1 or the word all if you want to install the Administration Console.

11. The installation procedure displays the name of the component you are installing, and lists copyright information. For example, if you selected 1 in the initial component selection menu, and then selected 3, CORBA Java Servers, the procedure displays the following:

BEA WebLogic Enterprise CORBA Java Server
(sparc) Release 5.1
Copyright (c) 2000 BEA Systems, Inc.
All Rights Reserved.
BEA and WebLogic are trademarks of BEA Systems, Inc.
SSLplus is a trademark of Certicom Corporation, 2000.
BSAFE is a trademark of RSA Data Security, Inc., 2000.

12. Enter the target directory for the selected software. The following prompt is displayed:

Directory where WebLogic Enterprise files are to be installed [?,q]:

For example, you could enter the /usr/local/wledir directory.

Most server components can be installed in any directory whose file system has enough disk space to accommodate them. There may be one or more server components that have to be installed over an existing WebLogic Enterprise server component.
When you enter the directory name, the installation program verifies it by using the following criteria:

- The value entered must be a full directory specification.
- If the directory already exists, it must be writable by the user performing the installation.
- If the directory already exists, it must have its executable permission set for the user performing the installation.
- The name provided for the directory cannot be an existing file.

If the target directory name passes the verification check and does not exist, the installation program creates the directory.

13. The file system for the target directory is checked for available space. For example:

```
Determining if sufficient space is available ...  
272989 blocks are required  
16118268 blocks are available to /usr/local/wledir  
Using /usr/local/wledir as the WebLogic Enterprise base directory  
```

If enough disk space is available, the installation continues. If there is insufficient disk space, the installation returns to the prompt asking for the name of a directory.

14. If you entered 3 or all in the initial component selection menu, indicating that the installation would include the Administration Console, you are asked to choose between the following:

- Accepting default locations for files being installed
- Specifying nondefault pathnames for these files

For details on the options, see the section “Selecting Directories for the WebLogic Enterprise Files” on page 1-14.

Press the Enter key to accept the default locations, if desired. If you accept the default locations, the following prompt is displayed. (In this sample, the user entered /usr/local/wledir as the target directory.)

```
Creating /usr/local/wledir/udataobj/webgui  
Using /usr/local/wledir/udataobj/webgui as the BEA Administration Console document tree  
```
Directory where BEA Administration Console java applets are to be installed (default: /usr/local/wledir/udataobj/webgui/java) [?,q]:

15. Again, you have a choice. If you accept the default, the following prompt is displayed:

Creating /usr/local/wledir/udataobj/webgui/java
Using /usr/local/wledir/udataobj/webgui/java as the BEA Administration Console document tree

Directory where BEA Administration Console CGI programs are to be installed (default: /usr/local/wledir/udataobj/webgui/cgi-bin) [?,q]:

Press the Enter key to accept the default locations, if desired.

16. If you accept the default, the following prompt is displayed:

Creating /usr/local/wledir/udataobj/webgui/cgi-bin
Using /usr/local/wledir/udataobj/webgui/cgi-bin as the BEA Administration Console CGI directory

Web server client prefix for CGI directory. /cgi-bin is a good choice for most web servers. (default: /cgi-bin) [?,q]:

Press the Enter key to accept the default locations, if desired.

17. If you accept the default, the following prompt is displayed:

Using /cgi-bin as the BEA Administration Console CGI prefix

18. At this point, the installation program proceeds to install the WebLogic Enterprise files.

19. After the installation of the WebLogic Enterprise files is completed, the following text and prompt is displayed for each server component you have chosen to install:

... finished
...
.
.
.
 Changing file permissions...
... finished
Processing default license file...
... finished
Install tlisten password? [y/n]:

20. If you want to specify a tlisten password, enter y; otherwise, enter n. For information about the tlisten password and instructions for setting it, see the section “Selecting an Administrative Password” on page 1-17. If you enter y, the following prompt is displayed:

Please enter the tlisten password:

21. Enter the tlisten password. The following prompt is displayed for each server component that you have chosen to install:

Please verify the password:

22. Enter the tlisten password again. The following prompt is displayed:

   tlistpwd: INFO: Password appended to file "/usr/local/wledir/udataobj/tlisten.pw".

Verifying installation...
... Installation successful!

If your license file is accessible, you may install it now.

Install license file? [y/n]:

23. If you want to install the WebLogic Enterprise software license now, enter y; otherwise, enter n and install the license later. If you enter y, the following prompt is displayed:

   To terminate the license update at any time press the interrupt key, typically <del>, <break>, or <ctrl+c>.

   Directory containing source license text file [?,q]:

   Warning: Pressing one of the interrupt keys mentioned in the prompt terminates the installation, not just the license update.

24. Insert the license diskette, which is shipped in the WebLogic Enterprise software box, in the diskette reader on your machine, mount the disk (if necessary), copy the lic.txt file to a system directory, and enter the location of the lic.txt file at the prompt. For example, if you copy the lic.txt file to /usr, enter /usr. The following prompt is displayed:

   Using /usr/lic.txt to copy license information.

   Updating /usr/local/wledir/udataobj/lic.txt with license information.

   Please don’t forget to fill out and send in your registration card.
25. After the installation is completed, unmount the CD file system and remove the CD from the reader. For platform-specific instructions for unmounting the CD, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”

### Installing the T-Engine Product License After You Install the WebLogic Enterprise Software

If you elected not to install the T-Engine product license when you installed the WebLogic Enterprise software, you can install the T-Engine license using the BEA License Utility.

**Note:** Your T-Engine product license is on a 3.5-inch diskette that is either attached to the outside of the WebLogic Enterprise product box, or mailed to you. For general information about licenses, see the section “Checking the WebLogic Enterprise Product Boxes” on page 1-2.

To install the T-Engine license, complete the following steps:

1. Insert the license diskette WebLogic Enterprise in the diskette reader on your machine, mount the disk (if necessary), and copy the lic.txt file to a directory of your choice, but not to the wledir directory or any of its subdirectories.

2. Change to the bin directory where you installed the WebLogic Enterprise software. For example:
   ```
   cd /usr/local/wledir/bin
   ```

3. Enter `sh ./lic.sh`.

   The following prompt is displayed:
   ```
   To terminate the license update at any time press the interrupt key, typically <del>, <break>, or <ctrl+c>.
   Directory containing source license text file [?,q] :
   ```

4. Enter the name of the directory that contains the lic.txt file (for example, /kits/license). The following prompt is displayed:

   ```
   Using /kits/license/lic.txt to copy license information.
   Directory where WebLogic Enterprise files are installed. [?,q] :
   ```
5. Enter `/usr/local/wledir` or the name of the directory where you installed the WebLogic Enterprise software. The following prompt is displayed:

```
Updating /usr/local/wledir/udataobj/lic.txt with license information.
```

**If you Installed WebLogic Enterprise on Solaris**

If you installed the WebLogic Enterprise software on a Solaris system, there is a way to tell which version of WebLogic Enterprise (Standard Mode or Compatibility Mode) was installed. The `/udataobj` directory under the installed WebLogic Enterprise directory contains either a `CompatibilityMode.txt` file or a `StandardMode.txt` file. The file has one line that states: "You have installed components for Sun Solaris v<OS-version> <mode-type>.

**Running Simpapp to Verify the WebLogic Enterprise CORBA C++ Software Installation**

To verify that you have successfully installed the WebLogic Enterprise CORBA C++ client and server software, run the `simpapp` application. This "simple application" is a WebLogic Enterprise client/server application that converts strings to uppercase and lowercase letters.

**Note:** This section assumes you installed all WebLogic Enterprise server components, or one of the CORBA server components. If you installed only the J2EE server component, see the next section, “Running Simpapp to Verify the WebLogic Enterprise CORBA Java Software Installation” on page 3-15, for information about running an EJB sample to verify the installation.

Before running the sample application, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets” for important information about prerequisite software.

To run `simpapp`, complete the following steps:
1. Make sure that the directory in which you installed the WebLogic Enterprise software is set in the environment variable TUXDIR. For example, if you installed the software in the default directory, enter the following to set the TUXDIR environment variable: TUXDIR=/usr/local/wledir; export TUXDIR.

2. Create a directory under wledir and copy the content of the simpapp directory to it.

Notes: If you installed all WebLogic Enterprise servers or the CORBA C++ server component in the default directory, a C++ simpapp directory is located at /usr/local/wledir/samples/corba/simpapp.

3. Change (cd) to the copy directory.

4. To change the permissions on all the files to allow full access, enter:
   chmod 777 *

5. Ensure that make is in your path.

6. To run simpapp automatically, enter ./runme.ksh. The simpapp application runs and prints the following messages:
   
   Testing simpapp
   cleaned up
   prepared
   built
   loaded ubb
   booted
   ran
   shutdown
   saved results
   PASSED

7. To run simpapp manually to observe the processes starting and stopping, do the following:
   
a. Enter KSH.

b. Enter . ./results/setenv.ksh.

c. Enter tmloadcf -y results/ubb.

d. Enter tmboot -y. The application starts several processes.

e. Enter ./simple_client. The prompt String? is displayed.
f. Enter a word in lowercase letters. The application converts the word to uppercase and then to lowercase letters and displays the results.

g. Enter `tmshutdown -y`. The application shuts down the processes.

8. To restore the directory to its original state, enter:

   a. `./results/setenv.ksh`

   b. `make -f makefile.mk clean`

---

**Running Simpapp to Verify the WebLogic Enterprise CORBA Java Software Installation**

To verify that you have successfully installed the WebLogic Enterprise CORBA Java client and server software, run the `simpapp_java` application. This “simple application” is a WebLogic Enterprise client/server application that converts strings to uppercase and lowercase letters.

**Notes:** This section assumes you installed all WebLogic Enterprise server components, or the CORBA Java server components. If you installed only the J2EE server component, see the next section, “Running Simpapp to Verify the WebLogic Enterprise CORBA Java Software Installation” on page 3-15, for information about running an EJB sample to verify the installation.

If you have installed the CORBA Java server components, the CORBA C++ components are also automatically installed. Therefore, if running `simpapp_java` as described in this section is successful, it also verifies that the CORBA C++ components were installed correctly.

Before running the sample application, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets” for important information about prerequisite software. For example, on NT systems, you must first download and install the Java 2 SDK version 1.2.2.

Before running the sample application, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets” for important information about prerequisite software.

To run `simpapp_java`, complete the following steps:
1. Make sure that the directory in which you installed the WebLogic Enterprise software is set in the environment variable TUXDIR. For example, if you installed the software in the default directory, enter the following to set the TUXDIR environment variable: TUXDIR=/usr/local/wledir; export TUXDIR.

2. Create a directory under wledir and copy the content of the simpapp directory to it.

Notes: If you installed all WebLogic Enterprise servers or the CORBA Java server component in the default directory, a simpapp_java directory is located at /usr/local/wledir/samples/corba/simpapp_java.

3. Change (cd) to the copy directory.

4. To change the permissions on all the files to allow full access, enter:
   chmod 777 *

5. Ensure that make is in your path.

6. To run simpapp automatically, enter ./runme.ksh. The simpapp_java application runs and prints the following messages:

```
Testing simpapp
   cleaned up
   prepared
   built
   loaded ubb
   booted
   ran
   shutdown
   saved results
   PASSED
```

7. To run simpapp_java manually to observe the processes starting and stopping, do the following:
   a. Enter KSH.
   b. Enter . ./results/setenv.ksh.
   c. Enter tmloadcf -y results/ubb.
   d. Enter tmboot -y. The application starts several processes.
   e. Enter ./simple_client. The prompt String? is displayed.
f. Enter a word in lowercase letters. The application converts the word to uppercase and then to lowercase letters and displays the results.

g. Enter `tmshutdown -y`. The application shuts down the processes.

8. To restore the directory to its original state, enter:

   a. `./results/setenv.ksh`
   
   b. `make -f makefile.mk clean`

### Running a Basic EJB Sample to Verify the WebLogic Enterprise J2EE Software Installation

If you installed only the WebLogic Enterprise J2EE server component, you can run the stateless session EJB sample provided by the WebLogic Enterprise software to verify the installation.

Before running the sample application, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets” for important information about prerequisite software. For example, on NT systems, you must first download and install the Java 2 SDK version 1.2.2.

This sample demonstrates the usage of a stateless session EJB that uses a simple stock trader application. This sample demonstrates how the client must maintain any persistent state—such as the change in the cash account—across repeated calls to the session EJB. All the logic for the balance is encapsulated in the client, where all the persistence is provided by the container and the logic is maintained in the EJB.

The EJB in this sample provides basic trading methods, such as buying and selling stocks. Because there are no persistent stores involved in this sample, all the stock data are set in the deployment descriptor of the EJB as environment properties. The container supplies the data to the EJB through the JNDI lookup operation.

To build the EJB sample, complete the following steps:

1. Ensure that the directory in which you installed WebLogic Enterprise is set in the environment variable `TUXDIR`. Also make sure to set the `JAVA_HOME` environment variable to the directory path where you installed the Java 2 SDK software. For example:
JAVA_HOME = /usr/local/JDK1.2.2

2. Make a copy of the $TUXDIR/samples/j2ee.ejb directory into a working directory.

3. Change the directory to the working directory.

4. Change the permissions on all the files to give them write-access. For example:
   `prompt>chmod -R +w *`
   Change the permission of the runme.ksh file to give it execute permission, as in the following command:
   `prompt>chmod +x runme.ksh`

5. Run the JavaServer version of the sample automatically by entering the runme command:
   `prompt> . ./runme.ksh basic statelessSession`

6. A number of messages are displayed, along with information about whether the build procedure was successful. The sample is built, the servers are booted, and the client is run once.

After you have executed the runme command, you can run the samples manually if you like.

To run the samples manually:

1. Change the current directory to the basic/statelessSession directory.

2. Ensure that your environment is set correctly by entering the following command:
   `prompt> . ./setenv.ksh`

3. Boot the server, run the client, and shut down the server by entering the following commands:
   `prompt> tmboot -y`
   `prompt> . ./run_client.ksh`
   `prompt> tmshutdown -y`

To restore the sample application directory to its original state:

1. Change to the work directory.
2. Enter the following command, where \texttt{TUXDIR} is the directory in which you installed the WebLogic Enterprise software:

\begin{verbatim}
prompt>. \$TUXDIR/samples/j2ee.ejb/clean.ksh
\end{verbatim}

For more information about the stateless session EJB sample application, see the \textit{Guide to the EJB Sample Applications} in the WebLogic Enterprise online documentation.

---

\textbf{Removing (Uninstalling) the WebLogic Enterprise Software from Your System}

To remove the software from your system, use the following procedure. This procedure also removes the WebLogic Enterprise Encryption Package software, if present on your system.

1. Log on as the WebLogic Enterprise administrator or superuser.

2. Make sure that no BEA Tuxedo or WebLogic Enterprise client or server applications are running. Use \texttt{tmshutdown} to shut down all WebLogic Enterprise applications.

3. Enter the following command:

\begin{verbatim}
prompt> rm -rf wledir
\end{verbatim}

where \texttt{wledir} is the WebLogic Enterprise base directory.
Now that you have successfully installed the WebLogic Enterprise software, you must set up your machine and parts of the WLE software to prepare for developing or installing your application.

This topic includes the following sections:

- Configuring the WebLogic Enterprise System for Microsoft Windows 2000 and NT 4.0
- Setting Up Your Environment on UNIX Systems
- Editing a UBBCONFIG File
- Verifying IPC Requirements
- Creating the Universal Device List and TLOG
- Starting the tlisten Process on UNIX Systems
- Running buildtms and buildXAJS for WebLogic Enterprise Applications That Use XA Resource Managers
- Using the TYPE Parameter in the UBBCONFIG File
Configuring the WebLogic Enterprise System for Microsoft Windows 2000 and NT 4.0

In addition to the BEA Administration Console, the WebLogic Enterprise system provides a control panel applet that you can use to configure the WebLogic Enterprise machine for Microsoft Windows 2000 and NT 4.0.

This section describes how to use the applet to do the following:

- Access machines on a network by setting the Machines page.
- Modify environment variables on the Environment page.
- Direct system messages to the Microsoft Windows NT Event Log by setting the Logging page.
- Configure one or more tlisten processes to start automatically by setting the Listener page.
- Maximize system performance by tuning the IPC Resources page setting.
Accessing the Control Panel Applet

To access the control panel applet, proceed as follows:

1. Click Start—>Settings—>Control Panel. The Control Panel is displayed.
2. Click the BEA Administration icon. The BEA Administration screen is displayed.

![BEA Administration Screen]

**Accessing Machines on a Network**

To display the Machines page of the Control Panel, click the Machine tab.

The Machines page enables the WebLogic Enterprise system administrator to access any machine on the Microsoft Windows Network running Microsoft Windows 2000 or NT 4.0, where the administrator has login privileges. The system administrator can then set environment variables remotely; determine the location of BEA WebLogic Enterprise event logging; add, start, or remove tlisten services; and tune IPC resources. To access a remote machine, the administrator locates the machine on a network tree.
If you know a machine’s name, but not its work group, proceed as follows:

1. Click Select. The Enter Machine Name screen is displayed.

2. Enter the name of the remote machine on the Enter Machine Name window and click OK.

All subsequent actions on other folders in the control panel applet take place on the selected machine.

Modifying Environment Variables

To display the Environment page of the Control Panel, click the Environment tab.

Modifying WebLogic Enterprise environment variables is almost identical to modifying Microsoft Windows 2000 or NT 4.0 environment variables. The Variable field (see Figure 4-1) contains a list of the most commonly used WebLogic Enterprise environment variables.

To modify the variables:

1. To add or edit a variable, select the variable, enter its value in the Value field, and click Set.

2. To delete a variable, select the variable you want to delete, and click Delete.

3. Click OK or Apply to save any changes.
Directing WebLogic Enterprise Messages to the Windows Event Log

To display the Logging page (Figure 4-2) of the Control Panel, click the Logging tab.

You can set the Logging page to direct WebLogic Enterprise system messages to the Event Log for Windows 2000 or NT 4.0. You can select the Logging option or the traditional user log (Disk File), or both. If you want traditional user log (ULOG) messages, select the directory into which ULOG messages will be written, as well as the prefix for the filename. The default prefix is ULOG, and the default filename is ULOG.<mmddyy>.
Figure 4-2  WebLogic Enterprise Software for Microsoft Windows NT Logging Control Panel
WebLogic Enterprise T-Engine Postinstallation Considerations

To view Event Log entries, click Start—>Programs—>Administrative Tools—>Event Viewer. The Event Viewer window is displayed.

<table>
<thead>
<tr>
<th>Date</th>
<th>Time</th>
<th>Source</th>
<th>Category</th>
<th>Event</th>
<th>User</th>
<th>Computer</th>
</tr>
</thead>
<tbody>
<tr>
<td>5/14/98</td>
<td>2:25:06 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/14/98</td>
<td>2:25:06 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:25:06 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:25:06 PM</td>
<td>EventLog</td>
<td>None</td>
<td>0</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>EventLog</td>
<td>None</td>
<td>6005</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>3:34:41 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>EventLog</td>
<td>None</td>
<td>6005</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
<tr>
<td>5/13/98</td>
<td>2:33:18 PM</td>
<td>E80x</td>
<td>None</td>
<td>3</td>
<td>N/A</td>
<td>PDMZ1</td>
</tr>
</tbody>
</table>

Configuring tlisten Processes to Start Automatically

To display the Listener page (Figure 4-3) of the Control Panel, click the Listener tab.

You can configure one or more tlisten processes to start automatically when you boot your machine. To configure tlisten processes, proceed as follows:

1. On the Listener page, enter a port number in the Port Number field and click Add to add the service to the list.

2. After you click OK or Apply and reopen the control panel, you can start or stop tlisten services from the Listener page (see Figure 4-3). You can also use the Microsoft Windows NT control panel to start or stop a tlisten service or to configure the service to start automatically.
You can use the tlisten process to perform administrative actions in a server application across multiple machines. You must start the tlisten process on each machine before running the server application. Generally, you need one tlisten process for each server application running on the machine.

**Maximizing System Performance**

To display the IPC Resources page (Figure 4-4) of the Control Panel, click the IPC Resources tab.
The WebLogic Enterprise software for Microsoft Windows 2000 and NT 4.0 systems provides you with BEA Tuxedo IPC Helper (TUXIPC), an interprocess communication subsystem, that is installed with the product. On most machines, IPC Helper runs as installed; however, you can use the IPC Resources page of the control panel applet to tune the TUXIPC subsystem and maximize performance.

Figure 4-4  WebLogic Enterprise Software for Microsoft Windows 2000 and NT 4.0 IPC Resources Control Panel
With the IPC Resources control panel, you can set a variety of IPC resources. To define IPC settings for your WebLogic Enterprise machine, proceed as follows:

1. In the Current Resource Default box, click the Use Default IPC Settings check box to clear it.

2. Click the insert box.

3. Enter the name of your machine and press Enter.

4. Click the fields next to the IPC resources you want to set, enter the desired values, and click Apply. Clicking Apply saves the changes in the Registry Table. You must then stop and then restart the Tuxedo IPC Helper for the changes to take effect.

5. Click OK to close the Control Panel.

You can view the performance of a running WebLogic Enterprise server application on the Windows 2000 or NT 4.0 Performance Monitor. An example is shown in Figure 4-5.

For example, to start the Performance Monitor on an NT 4.0 system, click Start—>Programs—>Administration Tools—>Performance Monitor on the NT taskbar. The Performance Monitor screen is displayed.
Setting Up Your Environment on UNIX Systems

On a UNIX system, before you can invoke WebLogic Enterprise system commands, you need to set several environment variables. The Bourne shell script `wle.env`, located in the base directory you specified at installation time, serves as a model for setting these variables.

The following examples assume that you are using the Bourne shell:

- **TUXDIR** contains the full pathname of the directory in which you installed the WebLogic Enterprise software. For example, if you installed the WebLogic Enterprise software in `/var/opt/WLEDIR`, enter the following:
TUXDIR=/var/opt/WLEDIR; export TUXDIR

- **PATH** is the search path for commands. Include $TUXDIR/bin in your path. For example:
  
  PATH=$PATH:$TUXDIR/bin; export PATH

- **LD_LIBRARY_PATH** (on Solaris systems), **SHLIB_PATH** (on HP-UX systems), and **LIBPATH** (on IBM AIX systems) name the search path for dynamic shared libraries. These environment variables are needed only on systems that support dynamic shared libraries. Append $TUXDIR/lib to your existing library path. For example, on Solaris systems, set the path variable as follows:
  
  LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$TUXDIR/lib; export LD_LIBRARY_PATH

- **TUXCONFIG** contains the full pathname of the binary configuration file of a specific WebLogic Enterprise server application. Several WebLogic Enterprise system commands require TUXCONFIG to be set appropriately. For example, if your application’s binary configuration file is located in /var/opt/wleapp1/tuxconfig, set and export TUXCONFIG as follows:

  TUXCONFIG=/var/opt/wleapp1/tuxconfig; export TUXCONFIG

---

**Editing a UBBCONFIG File**

Each WebLogic Enterprise machine has a configuration file, commonly called the **UBBCONFIG** file, which specifies the system parameters that are dependent on the installation. Typically, the configuration file has a name that begins with **ubb** and ends with something mnemonic, such as **ubbsimple**. Usually, you must edit this file before you can boot the application.

As an example, Listing 4-1 shows the configuration file from the University sample applications. This file, **Ubb_bnt**, is delivered with the WebLogic Enterprise software and is located in **WLEDIR/samples/corba/university** (for UNIX systems) or **WLEDIR\samples\corba\university** (for Microsoft Windows systems).
To edit the configuration file for your application, replace the strings provided for the following values:

**IPCKEY**

<machine_name>

**APPDIR**

**TUXCONFIG**

**TUXDIR**

These values are highlighted as **boldface** text in Listing 4-1, “University Samples UBBCONFIG File,” on page 4-15. The values you need to provide are as follows:

**IPCKEY**

A numeric key that identifies the shared memory segment where the structures used by your application are located. The value must be greater than 32768 and less than 262143.

**machine_name**

The node name of the machine. To obtain the node name on a UNIX system, enter the `uname -n` command. If you are using a Microsoft Windows NT system and you do not know the node name of your machine, contact your system administrator. In the University sample application shown in Listing 4-1, “University Samples UBBCONFIG File,” on page 4-15, the machine name is SRV.

**APPDIR = string_value**

APPDIR refers to directories in which application and administrative servers will be booted. The `string_value` is the absolute pathname of that directory, optionally followed by a colon-separated list of other directory pathnames, on the machine being defined.

**TUXCONFIG = string_value**

TUXCONFIG is the binary version of the UBBCONFIG file, produced by `tmloadcf(1)`. The `string_value` is the absolute pathname of the file or device of the TUXCONFIG file.

**TUXDIR = string_value**

Names the base directory of the WebLogic Enterprise software. It must be an absolute pathname.

If you need to look up other values when editing your configuration file, the complete syntax can be found on the `ubbconfig(5)` reference page in the *BEA Tuxedo Reference* that is included in the WebLogic Enterprise online documentation.
Note: The configuration file must be edited before you use the `tmloadcf(1)` command to verify the IPC requirements; otherwise, the `tmloadcf(1)` command fails with syntax errors. For instructions on how to determine IPC requirements, see the section “Verifying IPC Requirements” on page 4-17.

Listing 4-1 University Samples UBBCONFIG File

```bash
# RESOURCES
IPKEY    55432
DOMAINID university
MASTER    SITE1
MODEL     SHM
LDBAL     N

# MACHINES
# Specify the name of your server machine
SRV

   LMID = SITE1
# Pathname of your copy of this sample application.
# Must match "APPDIR" in "setenv.cmd"
"APPDIR = d:\wlework\checkinbasic"
# Pathname of the tuxconfig file.
# Must match "TUXCONFIG" in "setenv.cmd"
"TUXCONFIG = "d:\wlework\checkin\basic\resultsb\tuxconfig"
# Pathname of the WebLogic Enterprise installation.
# Must match "TUXDIR" in "setenv.cmd"
```

Installation Guide 4-15
TUXDIR = "d:\wledir"
MAXWSCLIENTS = 10

#---------------------------------------------------------------
*GROUPS
SYS_GRP
LMID = SITE1
GRPNO = 1
ORA_GRP
LMID = SITE1
GRPNO = 2

#---------------------------------------------------------------
*SERVERS
DEFAULT:
RESTART = Y
MAXGEN = 5

# Start the Tuxedo System Event Broker. This event broker must
# be started before any servers providing the NameManager Service
#
TMSYSEVT
SRVGRP = SYS_GRP
SRVID = 1
# TMFFNAME is a BEA WebLogic Enterprise provided server that
# runs the
# object-transactional management services. This includes the
# NameManager and FactoryFinder services.
# The NameManager service is a BEA WebLogic Enterprise-specific
# service that maintains a mapping of application-supplied
# names to object references.
# Start the NameManager Service (-N option). This name manager
# is being started as a Master (-M option).
#
TMFFNAME
SRVGRP = SYS_GRP
SRVID = 2
CLOPT = "-A -- -N -M"
#
# Start a slave NameManager Service
#
TMFFNAME
SRVGRP = SYS_GRP
SRVID = 2
CLOPT = "-A -- -N"
#
# Start the FactoryFinder (-F) service
#
TMFFNAME
SRVGRP = SYS_GRP
SRVID = 3
CLOPT = "-A -- -F"
#
# Start the IR Server
Verifying IPC Requirements

The WebLogic Enterprise system uses Interprocess Communications (IPC) resources heavily. On many platforms, the default values for the parameters that control the size and quantity of the various IPC resources are below the minimums needed to run even a modest WebLogic Enterprise system application. Therefore, you may need to reset some of the parameters. After editing your configuration file, the next step is to determine whether the IPC resources suffice for the application.

To do this, enter the `tmloadcf(1)` command, specifying your edited configuration file as input:

```
tmloadcf -c ubbconfig
```
An example of the result for the University samples UBBCONFIG file is shown in Listing 4-2.

**Listing 4-2  Output Produced by tmloadcf -c**

```
Ipc sizing (minimum /T values only)...

Fixed Minimums Per Processor
SHMMIN: 1
SHMALL: 1
SEMMAP: SEMMNI

Variable Minimums Per Processor
SEMUME, SEMMNU, A

<table>
<thead>
<tr>
<th>Node</th>
<th>SEMMNS</th>
<th>SEMMSL</th>
<th>SEMMSL</th>
<th>SEMMNI</th>
<th>MSGMNI</th>
<th>MSGMAP</th>
<th>SHMSEG</th>
</tr>
</thead>
<tbody>
<tr>
<td>sftuxe</td>
<td>65</td>
<td>8</td>
<td>60</td>
<td>A + 1</td>
<td>28</td>
<td>56</td>
<td>403k</td>
</tr>
</tbody>
</table>
```

where 1 <= A <= 8.

The number of expected application clients per processor should be added to each MSGMNI value.

This output indicates that to run the University sample application, your system must have SEMUME, SEMMNU, and SEMMNS set to no less than 65. SEMMSL must be at least 8, and SEMMNI and SEMMAP must be at least 4 (assuming A is 3). MSGMNI must be at least 28, and MSGMAP must be at least 56. Finally, the product of SHMMAX and SHMSEG must be at least 403 KB.

The IPC values are dependent on the client or server application, and the numbers in this example reflect a very small configuration. If other client or server applications that use IPC resources are running on the same machine with a WebLogic Enterprise client or server application, the requirements of both applications must be satisfied. Also, every machine participating in an application must have sufficient IPC resources available.

If the current IPC resources are inadequate, you must increase the values of the associated IPC parameters. Additional information is available at these locations:

- For instructions on determining and changing the current IPC values for your platform, see the section “Tuning Parameters” for your platform in Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
Creating the Universal Device List and TLOG

Creating the Universal Device List and TLOG

The Universal Device List (UDL) is like a map of the WebLogic Enterprise file system. It is loaded into shared memory when the application is booted. The TLOG refers to a log in which information on transactions is kept until the transaction is completed.

Creating the UDL

To create the UDL, enter the following command before the application has been booted:

```
tmadmin -c
crdl -z config -b blocks
```

where `-z config` specifies the full pathname for the device where the UDL should be created, and `-b blocks` specifies the number of blocks to be allocated on the device. The value of `config` should match the value of the TLOGDEVICE parameter in the MACHINES section of the UBBCONFIG file.

**Note:** In general, the value that you supply for blocks should not be less than the value for TLOGSIZE. For example, if TLOGSIZE is specified as 200 blocks, specifying `-b 500` would not cause a degradation.
Creating the TLOG

Several parameters in the MACHINES section of the UBBCONFIG file are used to define a global transaction log (TLOG). The WebLogic Enterprise system administrator must manually create the device list entry for the TLOGDEVICE on each machine where a TLOG is needed. The device list entry can be created either before or after TUXCONFIG has been loaded, but it must be done before the machine is booted.

To create an entry in the UDL for the TLOGDEVICE, create the UDL as described previously on each machine that will be involved with global transactions. If the TLOGDEVICE is mirrored between two machines, it is not necessary to create an entry on the paired machine. The Bulletin Board Liaison (BBL) then initializes and opens the TLOG during the boot process.

Starting the tlisten Process on UNIX Systems

When used in a distributed environment, the WebLogic Enterprise system requires the capability to start, shut down, and administer processes on remote machines running WebLogic Enterprise servers. The tlisten(1) process provides this facility. Once tlisten is running, tmboot(1), for example, can start WebLogic Enterprise servers on remote machines.

The tlisten process is a generic listener process that operates with either of the two network interfaces: Sockets or TLI. It runs as a daemon process, and it can be started in several ways, as follows:

- By the UNIX system administrator as part of a UNIX initialization (boot) script
- By the WebLogic Enterprise system administrator as a cron job
- By the WebLogic Enterprise system administrator starting tlisten manually from the command line

In all cases, the same basic invocation syntax is used:

```bash
TUXDIR=WLEDIR; export TUXDIR
LD_LIBRARY_PATH=libpath:$LD_LIBRARY_PATH; export LD_LIBRARY_PATH
$TUXDIR/bin/tlisten -d devname -l nlsaddr -u appuid
```
Note: If your machine uses an environment variable other than \texttt{LD\_LIBRARY\_PATH} for the shared library path, specify that variable, instead.

The \texttt{-l} option is required. The \texttt{-d} option is not required. The value for \texttt{-d} represents the network device. The correct values for various platforms are shown in Table 4-1.

\begin{table}[h]
\centering
\caption{Network Devices for \texttt{tlisten}}
\begin{tabular}{|l|l|}
\hline
Platform & Device Name \\
\hline
Compaq Tru64 UNIX 4.0f & /dev/streams/xtiso/tcp \\
HP-UX 11.00 & /dev/null \\
IBM AIX 4.3.3 & /dev/null \\
SCO UnixWare 7.1.1 & /dev/tcp \\
Solaris 2.6 and Solaris 7 & /dev/tcp \\
\hline
\end{tabular}
\end{table}

The value for \texttt{-l} should be the same as that specified for the \texttt{NLSADDR} parameter in the \texttt{NETWORK} section of the configuration file. For information about determining the value of \texttt{NLSADDR}, see the \texttt{ubbconfig(5)} or \texttt{tlisten(1)} reference page in the \textit{BEA Tuxedo Reference} or \textit{Creating a Configuration File} in the WebLogic Enterprise online documentation.

Use the \texttt{-u} \texttt{appuid} option when the command is part of an installation script run by \texttt{root}. The value of \texttt{appuid} is the UID or login name of the WebLogic Enterprise system administrator; the numeric version is the same as the value of the UID parameter in the \texttt{RESOURCES} section of the configuration file. Therefore, even though the \texttt{tlisten} process is started by \texttt{root}, it runs with the effective UID of the owner of the WebLogic Enterprise installation. If \texttt{tlisten} is started by the WebLogic Enterprise system administrator either manually or as a \texttt{cron} job, the \texttt{-u} option is unnecessary, because the job is already owned by the correct account.
Running buildtms and buildXAJS for WebLogic Enterprise Applications That Use XA Resource Managers

For WebLogic Enterprise applications that use distributed transactions and XA-compliant resource managers, you must use the buildtms command to construct a transaction manager server load module. This requirement exists on UNIX and Windows NT systems. When the module has been created, it must reside in %TUXDIR%\bin on NT systems, or $TUXDIR/bin on UNIX systems.

Note: If you run the CORBA C++ University sample applications, or the WebLogic Enterprise Java Bankapp XA sample application, each sample’s makefile creates the TMS load module for you and calls it tms_ora.exe. Therefore, running buildtms as a separate step is necessary only if you do not plan to run any of these sample applications.

For information about the buildtms command with WebLogic Enterprise applications, see the buildtms(1) reference page in the BEA Tuxedo Reference that is included in the WebLogic Enterprise online documentation.

You also must use the buildXAJS command to build an XA resource manager that will be used with a JavaServerXA application group. See Using the JDBC Drivers in the WebLogic Enterprise online documentation.

Using the TYPE Parameter in the UBBCONFIG File

The TYPE parameter in the MACHINES section of the UBBCONFIG file specifies the invocation of the XDR (EXternal Data Representation) encode/decode routines when messages are passed between unlike machines. The term unlike applies even to
machines of the same type if the compiler on each machine is different. In such a case, give each machine a unique TYPE string to force the message to go through the encode/decode routines.

Internet Browser Requirements

The BEA Application Builder online help requires an Internet browser. When you run the online Help from within Application Builder, Netscape is required. If you open the online Help files directly from a browser, you can use Netscape or Microsoft Internet Explorer.
If you elected to install the Administration component, read this topic for important information about starting the BEA Administration Console.

This topic includes the following sections:

- System Requirements
- Setting Up Your Environment
- Starting the BEA Administration Console

System Requirements

This section lists the hardware, operating system resources, and browser you must provide to support the BEA Administration Console.

Platforms Supported

The BEA Administration Console runs on all server platforms that support the BEA WebLogic Enterprise 5.1 T-Engine server software. For a complete list of supported platforms, see the section “Supported Platforms” on page 6-2.
5 BEA Administration Console Startup

Hardware Requirements

In addition to the hardware and software requirements for installing WebLogic Enterprise client and server software on your particular platform (see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets”), the BEA Administration Console requires a color display device with the following capabilities:

- Resolution: 800 by 600 dpi or more is required; 1024 by 768 dpi or more is recommended.
- Colors: 256 colors or more is recommended.

Operating System Requirements

Given the broad outlines of an application design, you must verify the availability of operating system resources needed to support your application. Operating system resources include:

- System shared resources (IPC), which control the maximum message size and maximum queue length, among other things
- Resources governed by kernel parameters

For information about setting Microsoft Windows NT IPC parameters, see the section “Maximizing System Performance” on page 4-9.

For information about setting UNIX system IPC parameters, see the section “Verifying IPC Requirements” on page 4-17.

For more information about system tuning parameters for a particular platform, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
Browser Requirements

Table 5-1 shows the browsers that are supported for the BEA Administration Console.

Table 5-1  Supported Browsers for the BEA Administration Console

<table>
<thead>
<tr>
<th>Browsers Supported</th>
<th>Platforms Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Netscape 4.6.1 or</td>
<td>Microsoft Windows 2000, NT 4.0, Windows 98, Solaris 2.6,</td>
</tr>
<tr>
<td>Netscape 4.7</td>
<td>Solaris 7</td>
</tr>
<tr>
<td>Netscape 4.6.1</td>
<td>Compaq Tru64 Unix 4.0f and IBM AIX 4.3.3</td>
</tr>
<tr>
<td>Microsoft Internet</td>
<td>Microsoft Windows 2000, NT 4.0, and Windows 98</td>
</tr>
<tr>
<td>Explorer 5.0</td>
<td></td>
</tr>
</tbody>
</table>

On all supported UNIX platforms, Netscape 4.6.1 or better is supported if the BEA Administration console is configured for zero-bit encryption. On all supported Windows platforms, Netscape 4.6.1 or better, and Microsoft Internet Explorer 5.0, are supported if the Administration Console is configured for zero-bit encryption. Table 5-1 indicates the browser support if the Administration Console is configured for 40-bit, 56-bit, or 128-bit encryption.

An additional requirement is that all browsers must be running the browser vendor’s Java plug-in version 1.1 or later.

The Administration Console encryption level is set in its webgui.ini file with the ENCRYPTBITS parameter. The ENCRYPTBITS parameters can be set to 0, 40, 56, or 128. This parameter specifies the strength of encryption used in communication between the GUI applet and the Administration Console server. The default is 128 bit.

Setting Up Your Environment

To run the BEA Administration Console, you need to set up two servers:

- tuxwsvr
A Web server provided with the WebLogic Enterprise system software. (You are not required to use this server; you may, if you prefer, use your own commercial Web server.)

- **wlisten**
  A server required to administer the BEA Administration Console. It should be run on the master machine.

**Note:** You can use any machine that supports a Java-capable browser for performing WebLogic Enterprise system administration through the BEA Administration Console.

### Starting tuxwsvr

To start `tuxwsvr` on UNIX systems, enter:

```
$ tuxwsvr -l //machine:port -i \
$ {TUXDIR}/udataobj/tuxwsvr.ini
```

To start `tuxwsvr` on Microsoft Windows NT systems:

1. Open an MS-DOS window.
2. Enter `tuxwsvr -l//machine:port -i%TUXDIR%\udataobj\tuxwsvr.ini`

During installation, the `tuxwsvr.ini` file is created. Usually, you do not need to edit this file. Under certain circumstances, however, you may want to edit this file. For example, you may decide, after installation, to move your Java files to a nondefault directory. In that case, you would need to edit the pathnames in the initialization file appropriately. For details, see the `tuxwsvr(1)` reference page in the *BEA Tuxedo Reference*.

### Starting wlisten

To start `wlisten`, proceed as follows:
Starting the BEA Administration Console

1. Before starting wlisten, check the webgui.ini file (located in
   WLEDIR\udataobj\webgui for Microsoft Windows NT systems and in
   WLEDIR/udataobj/webgui for UNIX systems) to make sure that the default
   values assigned to the parameters during installation are appropriate. Otherwise,
   make the appropriate changes.

2. For example, on a machine called popeye, the default port assigned to wlisten is
   4003. To run wlisten with port 6060, edit the NADDR parameter line in the
   webgui.ini file, as follows:

   NADDR=//popeye:6060

   For details about other parameters in the webgui.ini file, see the wlisten(1)
   reference page in the BEA Tuxedo Reference.

3. Start the wlisten process:

   $ wlisten

Starting the BEA Administration Console

To start the BEA Administration Console, proceed as follows:

1. Start the browser.

2. Enter the following URL:

   http://<machine_name>:<port>/webguitop.html

   Use of this URL depends on the following assumptions:
   - You are using tuxwsvr with the file tuxwsvr.ini.
   - The webgui.ini file is in the default location, WLEDIR/udataobj/webgui.

   Note: If you are using a commercial browser on the default port (8080), you can
   use something like the following URL:

   http://ctomsn:8080/webguitop.html

   The BEA Administration Console entry page is displayed, including warranty
   and license notices.
3. To start the BEA Administration Console, click the prompt at the bottom of the screen. The Login window is displayed.
4. Enter your login name and password in the appropriate fields, and click LOGIN. The password must be one of the entries in the `tlisten.pw` file in the `WLEDIR/udataobj` directory. The main window of the BEA Administration Console is displayed.
Table 5-2 contains instructions for accessing additional information about the BEA Administration Console main window.

Table 5-2  Accessing Information About the BEA Administration Console Main Window

<table>
<thead>
<tr>
<th>If . . .</th>
<th>Then . . .</th>
</tr>
</thead>
<tbody>
<tr>
<td>The main window is displayed and you want to start working with the GUI.</td>
<td>See the “Tutorial” section in the BEA Administration Console online help.</td>
</tr>
<tr>
<td>The main window is displayed and you want to read a description of it.</td>
<td>See Chapter 2, “A Tour of the Main Window,” in the BEA Administration Console Online Help.</td>
</tr>
</tbody>
</table>
| The main window does not display and the Connect Failed error message is displayed. | 1. Enter the ps command to verify that the wlisten process is running.  
2. If wlisten is not running, open the webgui.ini file and, in the line “NADDR=//lcsoll:4003,” replace the port number (4003) with a valid port number.  
3. Enter wlisten again:  
   `$ wlisten -i \  
   WLEDIR/udataobj/webgui/webgui.ini`  
4. Check that the tuwsvr process is running at the port as described in the URL.  
5. Verify the password. It must match one of the entries in the tlisten.pw file in the WLEDIR/udataobj directory.  
6. Return to step 1 above. |

5. To exit the BEA Administration Console, click Domain—>Exit.

You may now start setting up your environment for your own application domain.
6 WebLogic Enterprise T-Engine Platform Data Sheets

This appendix contains detailed information about the platforms supported by the WebLogic Enterprise 5.1 T-Engine software.

This topic includes the following sections:

- Supported Platforms
- Compaq Tru64 UNIX on Alpha Systems
- HP-UX Version 11.0 (32-bit) on HP 9000 Series
- IBM AIX 4.3.3 on RS/6000
- Microsoft Windows 2000 and NT 4.0 (SP5) on Intel
- Microsoft Windows 98 and 95 on Intel
- SCO UnixWare 7.1.1
- Sun Microsystems Solaris 2.6 and Solaris 7 (32-Bit) SPARC

Each data sheet includes the following platform-specific information:

- A list of available WebLogic Enterprise packages
- Hardware, software, network, and disk space requirements
Instructions for mounting and unmounting the WebLogic Enterprise software CD

- Tuning parameters

**Supported Platforms**

Table 6-1 lists the supported platforms. Data sheets are provided for each platform.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Operating System</th>
<th>Release/Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compaq</td>
<td>Tru64 UNIX</td>
<td>4.0f (Alpha)</td>
</tr>
<tr>
<td>HP</td>
<td>HP-UX</td>
<td>11.00, 32-bit, plus patches B.11.00.B0315, for the HP 9000 Series</td>
</tr>
<tr>
<td>IBM</td>
<td>AIX</td>
<td>4.3.3.3 with APARS AIX patches (rev3) installed on the RS/6000</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Windows 2000</td>
<td>4.0 (Intel) plus Service Pack 5 (SP5)</td>
</tr>
<tr>
<td></td>
<td>Windows NT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows 98 (clients only)</td>
<td>Windows 98 on Intel</td>
</tr>
<tr>
<td></td>
<td>Windows 95 (clients only)</td>
<td>Service Pack 1 on Intel</td>
</tr>
<tr>
<td>SCO</td>
<td>UnixWare</td>
<td>7.1.1 (CORBA C++ and Tuxedo only)</td>
</tr>
<tr>
<td>Sun Microsystems</td>
<td>Solaris</td>
<td>Solaris 2.6 SPARC with patch 105591-07</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Solaris 7 SPARC (32-bit) with patch 106327-06</td>
</tr>
</tbody>
</table>
The following sections list requirements for the Compaq Tru64 UNIX platform for Alpha systems.

**Available BEA WebLogic Enterprise Version 5.1 Packages**

The WebLogic Enterprise 5.1 software components for the Compaq Tru64 UNIX platform for Alpha systems are as follows:

- **WebLogic Enterprise servers**, consisting of:
  - CORBA Java servers and J2EE servers
  - CORBA C++ servers
  - Tuxedo servers (always installed as a base component for any of the other WebLogic Enterprise servers)

- **WebLogic Enterprise clients**, consisting of:
  - CORBA Java clients, over IIOP
  - CORBA C++ clients, native and over IIOP
  - RMI/EJB clients, on IIOP
  - Tuxedo clients

- **WebLogic Enterprise Administration software**, consisting of the BEA Administration Console.

- **WebLogic Enterprise 56-bit Encryption Packages or 128-bit Encryption Packages software**, if you purchased this optional software. The Encryption Packages installation can occur only after you install the core WebLogic Enterprise 5.1 software.
Hardware Requirements

- Compaq Alpha processors that support Tru64 UNIX 4.0f
- 64 MB of RAM (minimum)
- Access to a compact disc (CD) reader

Software Requirements

Table 6-2 lists the Compaq Tru64 UNIX for Alpha systems software requirements.

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>Compaq Tru64 UNIX for Alpha Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>C/C++ compilers</td>
<td>Digital C Compiler 5.9-005</td>
</tr>
<tr>
<td></td>
<td>Digital C++ Compiler 6.1-029</td>
</tr>
<tr>
<td></td>
<td>These compilers are required only for the WebLogic Enterprise development environment</td>
</tr>
<tr>
<td>Tools for the Administration desktop</td>
<td>Netscape 4.6.1 is needed for the BEA Administration Console</td>
</tr>
<tr>
<td></td>
<td>Java 2 JRE 1.2.2-3 is needed for the BEA WebLogic EJB Deployer</td>
</tr>
<tr>
<td>Java 2 Software Development Kit (SDK) for the development environment</td>
<td>Java 2 SDK 1.2.2-3 for Tru64 UNIX</td>
</tr>
<tr>
<td>Non BEA CORBA Java client</td>
<td>Java 2 SDK 1.2 IDL ORB (run time)</td>
</tr>
<tr>
<td>Database for CORBA C++ applications</td>
<td>Oracle 7.3.4 or later</td>
</tr>
</tbody>
</table>
Additional Notes

- When using Oracle, Programmer/2000 Pro*C/C++ version 2.2.3.0.0 is required to build the WebLogic Enterprise University samples.
- EJB and RMI do not require C or C++.
- JDBC/XA users need a C compiler and linker to run the `buildtms` and `buildXAJS` commands.
- Java IDL users need a C preprocessor. On UNIX systems, the C preprocessor comes with the system.
- JNI users need a C or C++ compiler and linker.
- BEA Tuxedo users need a C or C++ compiler.
- CORBA C++ users need a C++ compiler and linker.
- For the BEA Administration Console, the table indicates browser support if the GUI is configured for 56-bit, or 128-bit encryption. If the GUI is configured for zero-bit encryption, Netscape 4.61 or later is supported.
- When the optional WebLogic Enterprise Encryption Package software has been installed:
  - The WebLogic Enterprise 5.1 ISL/ISH supports SSL 3.0 for IIOP connections.
The WebLogic Enterprise 5.1 CORBA C++, CORBA Java, and RMI clients support SSL 3.0, and SSL connectivity between these clients and the WebLogic Enterprise ISL/ISH has been certified.

Secure Sockets Layer (SSL) is not supported for Tuxedo client connections to the WebLogic Enterprise (Tuxedo) WSL/WSH. Tuxedo Link Level Encryption is available for encryption of these connections and encryption of connections between machines and domains.

Although WebLogic Server supports SSL connections from its clients, the WebLogic Enterprise Connectivity feature in WebLogic Server 4.5.1 does not support SSL in the connection pools between WebLogic Server and the WebLogic Enterprise ISL/ISH. However, SSL connectivity is supported between the J-Engine and the T-Engine in WebLogic Enterprise 5.1.

To support certificate-based authentication when using SSL, WebLogic Enterprise provides an LDAP-based certificate retrieval mechanism. This has been certified for use with the LDAP Directory server included with Netscape Enterprise Server.

**Network Requirements**

TCP/IP, using the SOCKETS network interface.
Disk Space Requirements

The disk space requirements for installation on Compaq Tru64 UNIX on Alpha systems depends on which components you select during the installation, and whether other WebLogic Enterprise 5.1 components have been installed previously. Use the estimates in Table 6-3 as guidelines. These requirements are approximate and have been rounded up to the nearest megabyte (MB).

Table 6-3  Compaq Tru64 UNIX for Alpha Systems Disk Space Requirements

<table>
<thead>
<tr>
<th>Components</th>
<th>Compaq Tru64 UNIX for Alpha Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>All WebLogic Enterprise servers, all WebLogic Enterprise clients, and the BEA Administration Console server software</td>
<td>87 MB</td>
</tr>
<tr>
<td>Servers only</td>
<td></td>
</tr>
<tr>
<td>All servers: 69 MB</td>
<td></td>
</tr>
<tr>
<td>Tuxedo server only: 22 MB</td>
<td></td>
</tr>
<tr>
<td>CORBA C++ server only: 43 MB</td>
<td></td>
</tr>
<tr>
<td>CORBA Java server only: 68 MB</td>
<td></td>
</tr>
<tr>
<td>J2EE server only: 69 MB</td>
<td></td>
</tr>
<tr>
<td>Clients only</td>
<td></td>
</tr>
<tr>
<td>All clients: 29 MB</td>
<td></td>
</tr>
<tr>
<td>Tuxedo client only: 8 MB</td>
<td></td>
</tr>
<tr>
<td>CORBA C++ client only: 20 MB</td>
<td></td>
</tr>
<tr>
<td>RMI/EJB client only: 6 MB</td>
<td></td>
</tr>
<tr>
<td>CORBA Java client only: 23 MB</td>
<td></td>
</tr>
<tr>
<td>Administration</td>
<td></td>
</tr>
<tr>
<td>BEA Administration Console: 12 MB</td>
<td></td>
</tr>
<tr>
<td>Encryption Packages, 56-bit or 128-bit</td>
<td></td>
</tr>
<tr>
<td>2 MB for LLE only on Tuxedo server or client system</td>
<td></td>
</tr>
<tr>
<td>4 MB for LLE and SSL</td>
<td></td>
</tr>
</tbody>
</table>
Mounting and Unmounting the CD

Mounting a CD requires the type CDFS. Because CDFS is a configurable kernel option, the following line must exist in the system configuration file:

```
options CDFS
```

If the system configuration file does not contain this line, modify the file and then rebuild the kernel.

To mount a CD, enter the following commands:

```
su
mkdir /cdrom
/usr/sbin/mount -r -t cdfs -o noversion /dev/rrz unit#c /cdrom
```

where `unit#c` is the unit number of your CD drive.

In almost all cases, the unit number of the CD drive on a new system is 4 (that is, `/dev/rrz4c`). However, to ensure that you have the correct unit number of the drive, enter the following:

```
su
file /dev/rrz*c
```

The output identifies the CD drive as an RRD disk. The unit number of the drive is in the left column. For example:

```
/dev/rrz4c: character special (8/4098) SCSI #0 RRD43 disk #32 (SCSI ID #4)
```

To unmount the CD, enter the following command:

```
umount /cdrom
```

where `cdrom` is the mounting point.

Tuning Parameters

You probably need to reconfigure the Tru64 UNIX kernel before running WebLogic Enterprise software, because the default values of some tuning parameters are too low.

To adjust the tuning parameters, proceed as follows:
1. Determine whether the current values are adequate.
   For instructions about determining whether the current tuning parameter values
   are adequate, refer to “Verifying IPC Requirements” on page 5-17.

2. Reset the tuning parameters as necessary.
   For instructions about reconfiguring, rebuilding, and rebooting, see the following
documentation from Compaq: the `doconfig(8)` man page and the BEA
WebLogic Enterprise System Tuning and Performance Management manual.

Table 6-4 shows the default settings for the parameters and the settings used for the
University sample applications. Use these settings as a starting point; however, your
applications may require different settings.

**Note:** The parameters currently set on your system are located in
`/sys/conf/<systemname>`. To display the parameters, log in as root and
enter `/usr/bin/x11/dxkerneltuner` at the command prompt.

**Table 6-4 University Sample Applications Default Settings**

<table>
<thead>
<tr>
<th>Compaq Tru64 UNIX Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
<th>Setting for University Sample Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>semmsns</td>
<td>SEMMNS</td>
<td>60 (SEMMNI*2)</td>
<td></td>
</tr>
<tr>
<td>semmni</td>
<td>SEMMNI</td>
<td>10</td>
<td>16</td>
</tr>
<tr>
<td>semmsl</td>
<td>SEMMSL</td>
<td>25</td>
<td>25</td>
</tr>
<tr>
<td>semume</td>
<td>SEMUME</td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>semopm</td>
<td></td>
<td>10</td>
<td>10</td>
</tr>
<tr>
<td>semvmx</td>
<td></td>
<td>32767</td>
<td>32767</td>
</tr>
<tr>
<td>semaem</td>
<td></td>
<td>16384</td>
<td>16384</td>
</tr>
<tr>
<td>msgmni</td>
<td>MSGMNI</td>
<td>50</td>
<td>84</td>
</tr>
<tr>
<td>msgmax</td>
<td>MSGMAX</td>
<td>8192</td>
<td>8192</td>
</tr>
<tr>
<td>msgmnb</td>
<td>MSGMNB</td>
<td>16384</td>
<td>16384</td>
</tr>
<tr>
<td>msgtql</td>
<td>MSGTQL</td>
<td>40</td>
<td>40</td>
</tr>
</tbody>
</table>
To determine the current value of a tunable parameter, examine the kernel configuration file located in the `/sys/conf` directory. This file typically has the same name as the node (machine) name.

To change the value of a tunable parameter, follow the instructions on the `dmconfig` man page.

To specify the value of a parameter that was previously unspecified, add a line such as the following to the kernel configuration file:

```
semmni 256
```

where `semmni` is the name of the parameter and `256` is its value.

---

Table 6-4  University Sample Applications Default Settings (Continued)

<table>
<thead>
<tr>
<th>Compaq Tru64 UNIX Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
<th>Setting for University Sample Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>maxusers</td>
<td>maxusers</td>
<td>varies</td>
<td>32</td>
</tr>
<tr>
<td>maxproc</td>
<td>NPROC</td>
<td>20+8*maxusers</td>
<td>32-72 per user</td>
</tr>
<tr>
<td>maxuprc</td>
<td>MAXUP</td>
<td>64</td>
<td>(NPROC * 9) / 10</td>
</tr>
</tbody>
</table>
HP-UX Version 11.0 (32-bit) on HP 9000 Series

The following sections list requirements for the HP-UX platform.

WebLogic Enterprise 5.1 Components

The WebLogic Enterprise 5.1 software components for the HP-UX platform are as follows:

- WebLogic Enterprise servers, consisting of:
  - CORBA Java servers and J2EE servers
  - CORBA C++ servers
  - Tuxedo servers (always installed as a base component for any of the other WebLogic Enterprise servers)

- WebLogic Enterprise clients, consisting of:
  - CORBA Java clients, over IIOP
  - CORBA C++ clients, native and over IIOP
  - RMI/EJB clients, on IIOP
  - Tuxedo clients

- WebLogic Enterprise Administration software, consisting of the BEA Administration Console.

- WebLogic Enterprise 56-bit Encryption Packages or 128-bit Encryption Packages software, if you purchased this optional software. The Encryption Packages installation can occur only after you install the core WebLogic Enterprise 5.1 software.
Hardware Requirements

- HP 9000 Series
- 64 MB of RAM (minimum)
- Access to a compact disc (CD) reader

Software Requirements

Table 6-5 lists the HP-UX 11.0 (32-bit) software requirements.

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>HP-UX 11.0 (32-Bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>HP-UX 11.0 32-bit plus patches B.11.00.B0315. Java users must apply Java 2 patches, which are available at <a href="http://www.unixsolutions.hp.com/products/java/2_60_software_content.html">http://www.unixsolutions.hp.com/products/java/2_60_software_content.html</a></td>
</tr>
<tr>
<td>C compiler</td>
<td>HP C/ANSI compiler A.11.01.00. Required for development only</td>
</tr>
<tr>
<td>C++ compiler</td>
<td>HP C++ compiler A.03.13. Required for development only</td>
</tr>
<tr>
<td>Tools for the Administration desktop</td>
<td>Netscape 4.72 or later is needed for the BEA Administration Console</td>
</tr>
<tr>
<td></td>
<td>Java 2 JRE 1.2.2.03 is needed for the BEA WebLogic EJB Deployer</td>
</tr>
<tr>
<td>Java 2 Software Development Kit (SDK) for the development environment</td>
<td>Java 2 SDK 1.2.2.03 for HP-UX 11.00</td>
</tr>
<tr>
<td>Non-BEA CORBA Java client</td>
<td>Java 2 SDK 1.2 IDL ORB (run time)</td>
</tr>
</tbody>
</table>
When using Oracle, Programmer/2000 Pro*C/C++ version 2.2.3.0.0 is required to build the WebLogic Enterprise University samples.

EJB and RMI do not require C or C++.

JDBC/XA users need a C compiler and linker to run the builtms and buildXAJS commands.

Java IDL users need a C preprocessor. On UNIX systems, the C preprocessor comes with the system.

JNI users need a C or C++ compiler.

BEA Tuxedo users need a C or C++ compiler or linker.

CORBA C++ users need a C++ compiler and linker.

For the BEA Administration Console, the table indicates browser support if the GUI is configured for 40-bit, 56-bit, or 128-bit encryption. If the GUI is configured for zero-bit encryption, Netscape 4.61 or later is supported.

When the optional WebLogic Enterprise Encryption Package software has been installed:

### Table 6-5  HP-UX 11.0 (32-Bit) Software Requirements (Continued)

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>HP-UX 11.0 (32-Bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Database for CORBA C++ applications</td>
<td>Oracle 8.0.5 or later for HP-UX 11.0</td>
</tr>
<tr>
<td>Database for CORBA Java and J2EE applications</td>
<td>Oracle 8.0.5 or later for HP-UX 11.0</td>
</tr>
<tr>
<td>JDBC drivers</td>
<td>jdbcKona/Oracle 8.0.5 (Type 2), supports XA</td>
</tr>
<tr>
<td>SSL certificate authorities</td>
<td>Verisign, Netscape</td>
</tr>
<tr>
<td>LDAP directory servers</td>
<td>Netscape Enterprise Server</td>
</tr>
</tbody>
</table>
The WebLogic Enterprise 5.1 ISL/ISH supports SSL 3.0 for IIOP connections.

The WebLogic Enterprise 5.1 CORBA C++, CORBA Java, and RMI clients support SSL 3.0, and SSL connectivity between these clients and the WebLogic Enterprise ISL/ISH has been certified.

SSL is not supported for Tuxedo client connections to the WebLogic Enterprise (Tuxedo) WSL/WSH. Tuxedo Link Level Encryption is available for encryption of these connections and encryption of connections between machines and domains.

Although WebLogic Server supports SSL connections from its clients, the WebLogic Enterprise Connectivity feature in WLS 4.5.1 does not support SSL in the connection pools between WLS and the WebLogic Enterprise ISL/ISH. However, SSL connectivity is supported between the J-Engine and the T-Engine in WebLogic Enterprise 5.1.

To support certificate-based authentication when using SSL, WebLogic Enterprise provides an LDAP-based certificate retrieval mechanism. This has been certified for use with the LDAP Directory server included with Netscape Enterprise Server.

Network Requirements

TCP/IP using the SOCKETS network interface

Disk Space Requirements

The disk space requirements for installation on HP-UX 11.0 systems depends on which components you select during the installation, and whether other WebLogic Enterprise 5.1 components have been installed previously. Use the estimates in Table 6-6 as guidelines. These requirements are approximate and have been rounded up to the nearest megabyte (MB).
Mounting and Unmounting the CD

To mount a CD, enter the following commands:

```bash
su
mkdir /cdrom
mount -F cdfs -o cdcase /dev/dsk/cdrom_device /cdrom
```

where `cdrom_device` is listed in the output of the `ioscan -f -n` command.

---

**Table 6-6  HP-UX 11.0 Disk Space Requirements**

<table>
<thead>
<tr>
<th>Components</th>
<th>HP-UX 11.0 Disk Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>All WebLogic Enterprise servers, all WebLogic Enterprise clients, and the BEA Administration Console server software</td>
<td>89 MB</td>
</tr>
<tr>
<td>Servers only</td>
<td>All servers: 71 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo server only: 20 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ server only: 50 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA Java server only: 70 MB</td>
</tr>
<tr>
<td></td>
<td>J2EE server only: 71 MB</td>
</tr>
<tr>
<td>Clients only</td>
<td>All clients: 31 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo client only: 7 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ client only: 22 MB</td>
</tr>
<tr>
<td></td>
<td>RMI/EJB client only: 6 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA Java client only: 25 MB</td>
</tr>
<tr>
<td>Administration</td>
<td>The server components of the BEA Administration Console: 12 MB</td>
</tr>
<tr>
<td>Encryption Packages, 56-bit or 128-bit</td>
<td>2 MB for LLE only on Tuxedo server or client system</td>
</tr>
<tr>
<td></td>
<td>4 MB for LLE and SSL</td>
</tr>
</tbody>
</table>

---
To unmount the CD, enter the following command:

```
umount /cdrom
```

where `cdrom` is the mounting point.

### Tuning Parameters

You probably need to reconfigure the HP-UX kernel before running BEA WebLogic Enterprise software because the default values of some tuning parameters are too low.

To adjust the tuning parameters, proceed as follows:

1. Determine whether the current values are adequate.
   
   For instructions about determining whether the current tuning parameter values are adequate, refer to “Verifying IPC Requirements” on page 4-17.

2. Reset the tuning parameters as necessary.
   
   For instructions about reconfiguring HP-UX, see “Setting Up a System” in the *HP-UX System Administration Tasks Manual*.

Table 6-7 lists the default settings for the parameters and the settings used for the University sample applications. Use these settings as a starting point; however, your applications may require different settings.

The parameters currently set on your system are located in `/stand/build/tune.h`.

<table>
<thead>
<tr>
<th>HP-UX Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
<th>Setting for University Sample Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>shmmx</td>
<td>SHMMAX</td>
<td>67108864</td>
<td>0x40000000</td>
</tr>
<tr>
<td>shmseg</td>
<td>SHMSEG</td>
<td>12</td>
<td>32</td>
</tr>
<tr>
<td>shmmni</td>
<td>SHMMNI</td>
<td>100</td>
<td>512</td>
</tr>
<tr>
<td>semmns</td>
<td>SEMMNS</td>
<td>128</td>
<td>(SEMMNI*2)</td>
</tr>
<tr>
<td>semmni</td>
<td>SEMMNI</td>
<td>64</td>
<td>NPROC*5</td>
</tr>
<tr>
<td>semmap</td>
<td>SEMMA</td>
<td>semmni+2</td>
<td>1</td>
</tr>
</tbody>
</table>
### Table 6-7 University Sample Application Default Settings (Continued)

<table>
<thead>
<tr>
<th>HP-UX Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
<th>Setting for University Sample Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>semmnu</td>
<td>SEMMNU</td>
<td>30</td>
<td>(SEMMNI / 2)</td>
</tr>
<tr>
<td>semume</td>
<td>SEMUME</td>
<td>10</td>
<td>64</td>
</tr>
<tr>
<td>msgmni</td>
<td>MSGMNI</td>
<td>50</td>
<td>NPROC</td>
</tr>
<tr>
<td>msgmap</td>
<td>MSGMAP</td>
<td>2+msgtql</td>
<td>MSGTQL + 2</td>
</tr>
<tr>
<td>msgmax</td>
<td>MSGMAX</td>
<td>8192</td>
<td>32768</td>
</tr>
<tr>
<td>msgmnb</td>
<td>MSGMNB</td>
<td>16384</td>
<td>65535</td>
</tr>
<tr>
<td>msgssz</td>
<td>MSGSSZ</td>
<td>8</td>
<td>128</td>
</tr>
<tr>
<td>msgtql</td>
<td>MSGTQL</td>
<td>40</td>
<td>(NPROC * 10)</td>
</tr>
<tr>
<td>msgseg</td>
<td>MSGSEG</td>
<td>2048</td>
<td>(MSGTQL * 4)</td>
</tr>
<tr>
<td>maxusers</td>
<td>MAXUSERS</td>
<td>32</td>
<td>200</td>
</tr>
<tr>
<td>nproc</td>
<td>NPROC</td>
<td>20+8*maxusers</td>
<td>(MAXUSERS * 3) + 64</td>
</tr>
<tr>
<td>maxuprc</td>
<td>MAXUPRC</td>
<td>50</td>
<td>(NPROC * 9) / 10</td>
</tr>
<tr>
<td>maxfiles</td>
<td>NFILES</td>
<td>60</td>
<td>15 * NPROC + 2048</td>
</tr>
</tbody>
</table>
IBM AIX 4.3.3

The following sections list requirements for the IBM AIX 4.3.3 platform.

Available BEA WebLogic Enterprise Version 5.1 Packages

The WebLogic Enterprise 5.1 software components for the IBM AIX 4.3.3 platform are as follows:

- WebLogic Enterprise servers, consisting of:
  - CORBA Java servers and J2EE servers
  - CORBA C++ servers
  - Tuxedo servers (always installed as a base component for any of the other WebLogic Enterprise servers)

- WebLogic Enterprise clients, consisting of:
  - CORBA Java clients, over IIOP
  - CORBA C++ clients, native and over IIOP
  - RMI/EJB clients, on IIOP
  - Tuxedo clients

- WebLogic Enterprise Administration software, consisting of the BEA Administration Console.

- WebLogic Enterprise 56-bit Encryption Packages or 128-bit Encryption Packages software, if you purchased this optional software. The Encryption Packages installation can occur only after you install the core WebLogic Enterprise 5.1 software.
Hardware Requirements

- IBM RS/6000
- 64 MB of RAM (minimum)
- Access to a compact disc (CD) reader

Software Requirements

Table 6-8 lists the IBM AIX 4.3.3 software requirements.

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>IBM AIX 4.3.3 Platform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>4.3.3 with APARS AIX patches (rev3) installed</td>
</tr>
<tr>
<td>C/C++ compilers</td>
<td>C++ for AIX Compiler 3.6.4 C++</td>
</tr>
<tr>
<td></td>
<td>This compiler is required only for the WebLogic</td>
</tr>
<tr>
<td></td>
<td>Enterprise development environment</td>
</tr>
<tr>
<td>Tools for the Administration desktop</td>
<td>Netscape 4.6.1 or later is needed for the BEA</td>
</tr>
<tr>
<td></td>
<td>Administration Console</td>
</tr>
<tr>
<td></td>
<td>Java 2 JRE 1.2.2 is needed for the BEA WebLogic EJB Deployer</td>
</tr>
<tr>
<td>Java 2 Software Development Kit (SDK) for the development</td>
<td>IBM Developer Kit for AIX, version 1.2.2</td>
</tr>
<tr>
<td>environment</td>
<td></td>
</tr>
<tr>
<td>Non BEA CORBA Java client</td>
<td>Java 2 SDK 1.2 IDL ORB</td>
</tr>
<tr>
<td>Database for CORBA C++ applications and Tuxedo</td>
<td>Oracle 8.0.5 or later; supports XA</td>
</tr>
</tbody>
</table>
When using Oracle, Programmer/2000 Pro*C/C++ version 2.2.3.0.0 is required to build the WebLogic Enterprise University samples.

EJB and RMI do not require C or C++.

JDBC/XA users need a C compiler and linker to run the `buildtms` and `buildXAJS` commands.

Java IDL users need a C preprocessor. On UNIX systems, the C preprocessor comes with the system.

JNI users need a C or C++ compiler and linker.

BEA Tuxedo users need a C or C++ compiler or linker.

CORBA C++ users need a C++ compiler and linker.

For the BEA Administration Console, the table indicates browser support if the GUI is configured for 40-bit, 56-bit, or 128-bit encryption. If the GUI is configured for zero-bit encryption, Netscape 4.61 or later is supported.

When the optional WebLogic Enterprise Encryption Package software has been installed:

- The WebLogic Enterprise 5.1 ISL/ISH supports SSL 3.0 for IIOP connections.
The WebLogic Enterprise 5.1 CORBA C++, CORBA Java, and RMI clients support SSL 3.0, and SSL connectivity between these clients and the WebLogic Enterprise ISL/ISH has been certified.

Secure Sockets Layer (SSL) is not supported for Tuxedo client connections to the WebLogic Enterprise (Tuxedo) WSL/WSH. Tuxedo Link Level Encryption is available for encryption of these connections and encryption of connections between machines and domains.

Although WebLogic Server supports SSL connections from its clients, the WebLogic Enterprise Connectivity feature in WLS 4.5.1 does not support SSL in the connection pools between WLS and the WebLogic Enterprise ISL/ISH. However, SSL connectivity is supported between the J-Engine and the T-Engine in WebLogic Enterprise 5.1.

To support certificate-based authentication when using SSL, WebLogic Enterprise provides an LDAP-based certificate retrieval mechanism. This has been certified for use with the LDAP Directory server included with Netscape Enterprise Server.

Network Requirements

TCP/IP using the SOCKETS network interface.
Disk Space Requirements

The disk space requirements for installation on an IBM AIX 4.3.3 system depends on which components you select during the installation, and whether other WebLogic Enterprise 5.1 components have been installed previously. Use the estimates in Table 6-9 as guidelines. These requirements are approximate and have been rounded up to the nearest megabyte (MB).

**Table 6-9 IBM AIX 4.3.3 Disk Space Requirements**

<table>
<thead>
<tr>
<th>Components</th>
<th>IBM AIX 4.3.3</th>
</tr>
</thead>
<tbody>
<tr>
<td>All WebLogic Enterprise servers, all WebLogic Enterprise clients, and the BEA Administration Console server software</td>
<td>87 MB</td>
</tr>
</tbody>
</table>
| Servers only | All servers: 69 MB  
Tuxedo server only: 22 MB  
CORBA C++ server only: 43 MB  
CORBA Java server only: 68 MB  
J2EE server only: 69 MB |
| Clients only | All clients: 29 MB  
Tuxedo client only: 8 MB  
CORBA C++ client only: 20 MB  
RMI/EJB client only: 6 MB  
CORBA Java client only: 23 MB |
| Administration | BEA Administration Console: 12 MB |
| Encryption Packages, 56-bit or 128-bit | 2 MB for LLE only on Tuxedo server or client system  
4 MB for LLE and SSL |
Mounting and Unmounting the CD

To mount a CD, examine the file /etc/filesystems to determine whether there is a standard place in which to mount a CD. If there is, enter the mount command and specify the directory named in the /etc/filesystems entry.

For example, to mount a CD if /etc/filesystems contains an entry that specifies /cd as the mount point for CDs, enter:

```
su
/etc/mount /cd
```

If /etc/filesystems does not contain a CD entry, enter:

```
su
mkdir /cd
/etc/mount -v cdrfs -r cd_device /cd
```

where cd_device is the name of the CD device file, typically /dev/cd0.

Alternatively, you can use the System Management Interface Tool (SMIT) to perform the mount. To use SMIT, enter:

```
smit mount
```

To unmount the CD, enter the following command:

```
umount /cdrom
```

where cdrom is the mounting point.

Tuning Parameters

No IPC configuration is required for AIX Release 4.3.4. To change the value of a kernel tuning parameter (maxproc only), do the following:

1. Acquire superuser privileges.
2. Determine the values of all tuning parameters.
3. Change the parameter’s value.
4. Reboot the system.
Microsoft Windows 2000 and NT 4.0 (SP5) on Intel

The following sections list requirements for Microsoft Windows 2000 and NT 4.0 (SP5) on the Intel platform.

BEA WebLogic Enterprise Version 5.1 Components

The WebLogic Enterprise 5.1 software components for the Windows 2000 and NT 4.0 platforms are as follows:

- WebLogic Enterprise servers, consisting of:
  - CORBA Java servers and J2EE servers
  - CORBA C++ servers
  - Tuxedo servers (always installed as a base component for any of the other WebLogic Enterprise servers)

- WebLogic Enterprise clients, consisting of:
  - CORBA Java clients, over IIOP
  - CORBA C++ clients, native and over IIOP
  - RMI/EJB clients, on IIOP
  - Tuxedo clients
  - ActiveX clients

- WebLogic Enterprise Administration software, consisting of the BEA Administration Console.

- WebLogic Enterprise 56-bit Encryption Packages or 128-bit Encryption Packages software, if you purchased this optional software. The Encryption Packages installation can occur only after you install the core WebLogic Enterprise 5.1 software.
Hardware Requirements

- Pentium processor or better
- 64 MB of RAM (minimum)
- Access to a compact disc (CD) reader

Software Requirements

Table 6-10 lists the Windows 2000 and NT 4.0 SP5 on Intel software requirements.

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>Windows 2000 and NT 4.0 SP5 on Intel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compilers</td>
<td>Microsoft Visual C++ 6.0, SP2.</td>
</tr>
<tr>
<td></td>
<td>Microsoft Visual Basic 6.0. This is required only for client systems that run ActiveX client applications. It is not required for C++, Java, and VisiJava client systems or server-only systems.</td>
</tr>
<tr>
<td>Java 2 Software Development Kit (SDK) for the development environment</td>
<td>Java 2 SDK 1.2.2_001.</td>
</tr>
<tr>
<td></td>
<td>Java Hotspot Server VM 1.0.1 for Windows.</td>
</tr>
<tr>
<td>Tools for the Administration desktop</td>
<td>Netscape 4.61, Netscape 4.7, or Microsoft Internet Explorer 5.0 is needed for the BEA Administration Console.</td>
</tr>
<tr>
<td></td>
<td>JRE 1.2.2 is needed for the BEA WebLogic EJB Deployer.</td>
</tr>
<tr>
<td>Non-BEA CORBA Java clients</td>
<td>Java 2 SDK 1.2 IDL ORB (run time).</td>
</tr>
</tbody>
</table>
### Table 6-10 Windows 2000 and NT 4.0 SP5 on Intel Software Requirements

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>Windows 2000 and NT 4.0 SP5 on Intel</th>
</tr>
</thead>
<tbody>
<tr>
<td>Internet browser for BEA Application Builder Help</td>
<td>Netscape 4.0 or later. This browser is required only for the online Help used in the BEA Application Builder. This graphical user interface is installed on your system if you selected ActiveX Clients.</td>
</tr>
<tr>
<td>SSL certificate authorities</td>
<td>Verisign. Netscape.</td>
</tr>
<tr>
<td>Database for CORBA C++ applications</td>
<td>Oracle 7.3.4 or 8.1.5 or later on NT 4.0 systems.</td>
</tr>
<tr>
<td></td>
<td>Oracle 8.0.5 or later on Windows 2000 systems.</td>
</tr>
<tr>
<td>Database for CORBA Java and J2EE applications</td>
<td>Oracle 8.1.5, also known as Oracle 8i, or later on NT 4.0 systems.</td>
</tr>
<tr>
<td></td>
<td>Oracle 8.0.5 or later on Windows 2000 systems.</td>
</tr>
<tr>
<td>JDBC drivers</td>
<td>Without XA support:</td>
</tr>
<tr>
<td></td>
<td>jdbcKona/Oracle 7.3.4 (Type 2) on NT 4.0 systems.</td>
</tr>
<tr>
<td></td>
<td>jdbcKona/Oracle 8.0.5 (Type 2) on Windows 2000 systems.</td>
</tr>
<tr>
<td></td>
<td>With XA support:</td>
</tr>
<tr>
<td></td>
<td>WebLogic Enterprise JDBC/XA for Oracle 8.1.5, also known as Oracle 8i, on both NT and Windows 2000 systems.</td>
</tr>
<tr>
<td>SSL certificate authorities</td>
<td>Verisign. Netscape.</td>
</tr>
<tr>
<td>LDAP directory server</td>
<td>Netscape Enterprise Server.</td>
</tr>
</tbody>
</table>

### Additional Notes

- When using Oracle, Programmer/2000 Pro*C/C++ version 2.2.3.0.0 is required to build the WebLogic Enterprise University samples.
- EJB and RMI do not require C or C++.
JDBC/XA users need a C compiler and linker to run the `buildtms` and `buildXAJS` commands.

Java IDL users need a C preprocessor. On NT, Visual C++ is needed.

JNI users need a C or C++ compiler and linker.

BEA Tuxedo users need a C or C++ compiler.

CORBA C++ users need a C++ compiler and linker.

For the BEA Administration Console, the table indicates browser support if the GUI is configured for 40-bit, 56-bit, or 128-bit encryption. If the GUI is configured for zero-bit encryption, Netscape 4.61 or later, or Microsoft Internet Explorer 5.0, is supported.

When the optional WebLogic Enterprise Encryption Package software has been installed:

- The WebLogic Enterprise 5.1 ISL/ISH supports SSL 3.0 for IIOP connections.
- The WebLogic Enterprise 5.1 CORBA C++, CORBA Java, and RMI clients support SSL 3.0, and SSL connectivity between these clients and the WebLogic Enterprise ISL/ISH has been certified.
- SSL is not supported for Tuxedo client connections to the WebLogic Enterprise (Tuxedo) WSL/WSH. Tuxedo Link Level Encryption is available for encryption of these connections and encryption of connections between machines and domains.
- Although WebLogic Server supports SSL connections from its clients, the WebLogic Enterprise Connectivity feature in WLS 4.5.1 does not support SSL in the connection pools between WLS and the WebLogic Enterprise ISL/ISH. However, SSL connectivity is supported between the J-Engine and the T-Engine in WebLogic Enterprise 5.1.

To support certificate-based authentication when using SSL, WebLogic Enterprise provides an LDAP-based certificate retrieval mechanism. This has been certified for use with the LDAP Directory server included with Netscape Enterprise Server.
Network Requirements

TCP/IP provided by Microsoft Windows NT (32-bit Winsock).

Disk Space Requirements

The disk space requirements for installation on Windows 2000 and NT 4.0 SP4 (Intel) systems depends on which components you select during the installation, and whether other WebLogic Enterprise 5.1 components have been installed previously. Use the estimates in Table 6-11 as guidelines. These requirements are approximate and have been rounded up to the nearest megabyte (MB).

<table>
<thead>
<tr>
<th>Components</th>
<th>Windows 2000 and NT 4.0 SP5 (Intel) Disk Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>All WebLogic Enterprise servers, all WebLogic Enterprise clients, and the BEA Administration Console server software</td>
<td>83 MB</td>
</tr>
<tr>
<td>Servers only</td>
<td>All servers: 54 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo server only: 21 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ server only: 37 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA Java server only: 53 MB</td>
</tr>
<tr>
<td></td>
<td>J2EE server only: 54 MB</td>
</tr>
<tr>
<td>Clients only</td>
<td>All clients: 42 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo client only: 13 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ client only: 24 MB</td>
</tr>
<tr>
<td></td>
<td>RMI/EJB client only: 5 MB</td>
</tr>
<tr>
<td></td>
<td>ActiveX client only: 13 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA Java client only: 24 MB</td>
</tr>
<tr>
<td>Administration</td>
<td>BEA Administration Console: 11 MB</td>
</tr>
</tbody>
</table>
Tuning Parameters

You may need to reconfigure the parameters shown in Figure 4-4, “WebLogic Enterprise Software for Microsoft Windows 2000 and NT 4.0 IPC Resources Control Panel,” on page 4-10 before running the WebLogic Enterprise software. For instructions about reconfiguring the parameters, see “Maximizing System Performance” on page 4-9.

Microsoft Windows 98 and 95

The following sections list requirements for the Windows 98 and 95 platforms.

Available BEA WebLogic Enterprise Version 5.1 Packages

Only the WebLogic Enterprise client software components are supported:

- CORBA Java clients, over IIOP
- CORBA C++ clients, native and over IIOP
- RMI/EJB clients, on IIOP
- ActiveX clients

Note: On Windows 95 systems, the ActiveX Client software requires DCOM 1.1 and a patch from Microsoft. For details, see: http://www.microsoft.com/msdn/downloads/files/40comupd.htm.
Hardware Requirements

- Pentium processor or better
- 32 MB of RAM (minimum)
- Access to a compact disc (CD) reader

Software Requirements

Table 6-12 lists the Microsoft Windows 98 and 95 software requirements.

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>Microsoft Windows 98 and 95</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Microsoft Windows 95 plus Service Pack 1, or Microsoft Windows 98.</td>
</tr>
<tr>
<td>DCOM</td>
<td>On Windows 95 systems that are used as ActiveX clients, DCOM for Windows 95 version 1.1 is required, plus a patch from Microsoft. For details, see <a href="http://www.microsoft.com/msdn/downloads/files/40comupd.htm">http://www.microsoft.com/msdn/downloads/files/40comupd.htm</a>.</td>
</tr>
</tbody>
</table>
| Compilers for client development          | Microsoft Visual C++ 6.0, SP2.  
Microsoft Visual Basic 6.0. This is required only for client systems that run ActiveX client applications. It is not required for C++, Java, and VisiJava client systems. |
| Java 2 Software Development Kit (SDK) for the development environment | Java 2 SDK 1.2.2_001. |
TCP/IP provided by Microsoft Windows NT (32-bit Winsock).

Network Requirements

TCP/IP provided by Microsoft Windows NT (32-bit Winsock).
Disk Space Requirements

The following table lists the Windows 98 and 95 disk space requirements.

<table>
<thead>
<tr>
<th>Components</th>
<th>Windows 95 or 98 Disk Space</th>
</tr>
</thead>
<tbody>
<tr>
<td>Clients only</td>
<td>All clients: 42 MB</td>
</tr>
<tr>
<td>Tuxedo client only: 13 MB</td>
<td></td>
</tr>
<tr>
<td>CORBA C++ client only: 24 MB</td>
<td></td>
</tr>
<tr>
<td>RMI/EJB client only: 5 MB</td>
<td></td>
</tr>
<tr>
<td>ActiveX client only: 13 MB</td>
<td></td>
</tr>
<tr>
<td>CORBA Java client only: 24 MB</td>
<td></td>
</tr>
</tbody>
</table>
SCO UnixWare 7.1.1

The following sections list requirements for the SCO UnixWare 7.1.1 platform. On this platform, only the CORBA C++ and Tuxedo components in WebLogic Enterprise are supported. The WebLogic Enterprise Java components are not supported.

Available BEA WebLogic Enterprise Version 5.1 Packages

The WebLogic Enterprise 5.1 software components for the SCO UnixWare 7.1.1 platform are as follows:

- WebLogic Enterprise servers, consisting of:
  - CORBA C++ servers
  - Tuxedo servers (always installed as a base component for any of the other WebLogic Enterprise servers)

- WebLogic Enterprise clients, consisting of:
  - Tuxedo Workstation clients

- WebLogic Enterprise Administration software, consisting of the BEA Administration Console.

- WebLogic Enterprise 56-bit Encryption Packages or 128-bit Encryption Packages software, if you purchased this optional software. The Encryption Packages installation can occur only after you install the core WebLogic Enterprise 5.1 software.

Hardware Requirements

- Pentium processor or better
- 128 MB of RAM (minimum)
- Access to a compact disc (CD) reader
Software Requirements

Table 6-13 lists the SCO UnixWare 7.1.1 software requirements.

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>SCO UnixWare 7.1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>C/C++ compiler</td>
<td>MIPS Pro compilers (C/C++) 7.2.1, with the following patches: SG0002992, SG0002991, SG0003048, SG0003077, SG0003131, and SG0003139. This compiler is required only for the WebLogic Enterprise development environment.</td>
</tr>
<tr>
<td>Internet browsers for the BEA Administration Console</td>
<td>Netscape 4.6.1. Netscape 4.7.</td>
</tr>
<tr>
<td>Database for CORBA C++ applications</td>
<td>Oracle 8.0.5 or later.</td>
</tr>
<tr>
<td>SSL certificate authorities</td>
<td>Verisign. Netscape.</td>
</tr>
<tr>
<td>LDAP directory server</td>
<td>Netscape Enterprise Server.</td>
</tr>
</tbody>
</table>

Additional Notes

- When using Oracle, Programmer/2000 Pro*C/C++ version 2.2.3.0.0 is required to build the WebLogic Enterprise University samples.
- BEA Tuxedo users need a C or C++ compiler or linker.
- CORBA C++ users need a C++ compiler and linker.
- For the BEA Administration Console, the table indicates browser support if the GUI is configured for 40-bit, 56-bit, or 128-bit encryption. If the GUI is configured for zero-bit encryption, Netscape 4.61 or later is supported.
Network Requirements

TCP/IP using the TLI network interface.

Disk Space Requirements

The disk space requirements for installation on a SCO UnixWare 7.1.1 system depends on which components you select during the installation, and whether other WebLogic Enterprise 5.1 components have been installed previously. Use the estimates in Table 6-14 as guidelines. These requirements are approximate and have been rounded up to the nearest megabyte (MB).

Table 6-14  SCO UnixWare 7.1.1 Disk Space Requirements

<table>
<thead>
<tr>
<th>Components</th>
<th>SCO UnixWare 7.1.1</th>
</tr>
</thead>
<tbody>
<tr>
<td>All WebLogic Enterprise C++ and Tuxedo servers, all WebLogic Enterprise C++ and Tuxedo clients, and the BEA Administration Console server software</td>
<td>80 MB</td>
</tr>
<tr>
<td>Servers only</td>
<td>All servers: 44 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo server only: 22 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ server only: 43 MB</td>
</tr>
<tr>
<td>Clients only</td>
<td>All clients: 21 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo client only: 8 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ client only: 20 MB</td>
</tr>
<tr>
<td>Administration</td>
<td>BEA Administration Console: 12 MB</td>
</tr>
<tr>
<td>Encryption Packages, 56-bit or 128-bit</td>
<td>2 MB for LLE only on Tuxedo server or client system</td>
</tr>
<tr>
<td></td>
<td>4 MB for LLE and SSL</td>
</tr>
</tbody>
</table>
Mounting and Unmounting the CD

To mount a CD, execute the following from the command line:

```
su
mkdir /cdrom1
mount -F cdfs -r -o nmconv=m cd-device /cdrom1
```

To determine the value of `cd-device`, execute:

```
devattr cdrom1 bdevice
```

You can also mount a CD from the Desktop environment. From the UnixWare Desktop window, open the Disks-etc Folder Window by double-clicking on the Disks-etc icon. Then, from the Disks-etc window double-click on the cdrom1 icon, which mounts the CD and displays its contents in the `/cdrom1 Folder Window. When you close the `/cdrom1 Folder Window, the CD is automatically unmounted.

UnixWare 7 typically defines `/var/tmp` as an in-memory file system. Because `/var/tmp` is used by the install script to unspool packages from the CD you should set (and export) `TMPDIR` to point to a directory with enough free space to accommodate the unspooling.

To share (export) a CD so that it can be accessed as a shared file system using NFS, execute:

```
su
share -o ro mount-point
```

where `mount-point` is the full pathname of the directory where the CD is mounted. The directory `/cdrom1` is the standard mount point for a CD. After a CD has been shared and mounted, a remote system can access the CD by mounting it as an NFS file system. For example, on another UnixWare system the appropriate commands are:

```
su
mkdir /cdrom1
mount -F nfs server:/cdrom1 /cdrom1
```

where `server` is the node name of the UnixWare system with the directly-connected CD-ROM device.
Tuning Parameters

You may need to reconfigure the UnixWare kernel before running the WebLogic Enterprise system because the default values of some IPC parameters are too low. For instructions about reconfiguring, rebuilding, and rebooting UnixWare, see *UnixWare System Performance Administration* and the `idtune(1)` and `idbuild(1)` entries in a UNIX system reference manual. You may need to increase the parameters listed in Table 6-15.

Table 6-15  SCO UnixWare Tuning Parameters

<table>
<thead>
<tr>
<th>SCO UnixWare Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>shmmmax</td>
<td>SHMMAX</td>
<td>524288</td>
</tr>
<tr>
<td>shmseg</td>
<td>SHMSEG</td>
<td>6</td>
</tr>
<tr>
<td>shmmni</td>
<td>SHMMNI</td>
<td>100</td>
</tr>
<tr>
<td>semmns</td>
<td>SEMMNS</td>
<td>60</td>
</tr>
<tr>
<td>semmni</td>
<td>SEMMNI</td>
<td>10</td>
</tr>
<tr>
<td>semmsl</td>
<td>SEMMSL</td>
<td>25</td>
</tr>
<tr>
<td>semmap</td>
<td>SEMMA</td>
<td>10</td>
</tr>
<tr>
<td>semmmnu</td>
<td>SEMMNU</td>
<td>30</td>
</tr>
<tr>
<td>semume</td>
<td>SEMUME</td>
<td>10</td>
</tr>
<tr>
<td>msgmni</td>
<td>MSGMNI</td>
<td>50</td>
</tr>
<tr>
<td>msgmap</td>
<td>MSGMAP</td>
<td>100</td>
</tr>
<tr>
<td>msgmax</td>
<td>MSGMAX</td>
<td>2048</td>
</tr>
<tr>
<td>msgmnb</td>
<td>MSGMNB</td>
<td>4096</td>
</tr>
<tr>
<td>msgsz</td>
<td>MSGSSZ</td>
<td>8</td>
</tr>
<tr>
<td>msgtql</td>
<td>MSGTQL</td>
<td>40</td>
</tr>
<tr>
<td>msgseg</td>
<td>MSGSEG</td>
<td>1024</td>
</tr>
</tbody>
</table>
To change the value of a tunable parameter, complete the following procedure.

1. Acquire superuser privileges:
   ```
   su root
   ```

2. Run the `idtune` command to determine the current default value of the tunable parameter to be changed:
   ```
   /etc/conf/bin/idtune -g param
   ```
   The output of `idtune` lists current, default, minimum, and maximum valid values.

3. Change the parameter’s value:
   ```
   /etc/conf/bin/idtune param new-value
   ```

4. If `idtune` fails because `new-value` exceeds the maximum allowed value, you must increase the maximum. (The maximum values of some parameters are set unnecessarily low in some releases of UnixWare.) To do this, edit:
   ```
   /etc/conf/mtune.d/type
   ```
   where type is `shm`, `sem`, or `msg`. Change the maximum value (in the last column) to the desired maximum.

5. Rebuild the operating system and reboot:
   ```
   /etc/conf/bin/idbuild -B
   cd /
   shutdown -y -g0 -i6
   ```

6. Execute `idtune` again and supply the desired `new-value`.

7. Run `idbuild` to build a new kernel:
   ```
   /etc/conf/bin/idbuild -B
   cd /
   shutdown -y -g0 -i6
   ```

---

**Table 6-15  SCO UnixWare Tuning Parameters (Continued)**

<table>
<thead>
<tr>
<th>SCO UnixWare Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
</tr>
</thead>
<tbody>
<tr>
<td>nproc</td>
<td>NPROC</td>
<td>200</td>
</tr>
<tr>
<td>maxup</td>
<td>MAXUP</td>
<td>200</td>
</tr>
</tbody>
</table>
8. Reboot your system.

**Sun Microsystems Solaris 2.6 and Solaris 7 (32-Bit) SPARC**

The following sections list requirements for the Sun Microsystems Solaris platform.

**Available BEA WebLogic Enterprise Version 5.1 Packages**

The WebLogic Enterprise 5.1 software components for the Solaris 2.6 and Solaris 7 (32-bit) platforms are as follows:

- WebLogic Enterprise servers, consisting of:
  - CORBA Java servers and J2EE servers
  - CORBA C++ servers
  - Tuxedo servers (always installed as a base component for any of the other WebLogic Enterprise servers)

- WebLogic Enterprise clients, consisting of:
  - CORBA Java clients, over IIOP
  - CORBA C++ clients, native and over IIOP
  - RMI/EJB clients, on IIOP
  - Tuxedo clients

- WebLogic Enterprise Administration software, consisting of the BEA Administration Console.

- WebLogic Enterprise 56-bit Encryption Packages or 128-bit Encryption Packages software, if you purchased this optional software. The Encryption Packages installation can occur only after you install the core WebLogic Enterprise 5.1 software.
Hardware Requirements

- SPARC
- 64 MB of RAM (minimum)
- Access to a compact disc (CD) reader

Software Requirements

Table lists the Solaris SPARC 2.6 and Solaris SPARC 7 software requirements.

Table 6-16 Solaris SPARC 2.6 and Solaris SPARC 7 Software Requirements

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>Solaris SPARC 2.6 and Solaris SPARC 7 (32-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Operating system</td>
<td>Solaris SPARC 2.6 with patch 105591-07.</td>
</tr>
<tr>
<td></td>
<td>Solaris SPARC 7 (32-bit) with patch 106327-06.</td>
</tr>
<tr>
<td>C/C++ compilers</td>
<td>The following compilers are required on both Solaris 2.6 and Solaris 7, but only for a WebLogic Enterprise development environment. Sun Microsystems, Inc. Workshop Compiler C++ 5.0 Standard mode and Compatibility mode, plus patches 107289-03, 107357-05, 107311-09. While Sun's patches are cumulative, they are only cumulative for a given component, and each of these three patches pertains to a different component:</td>
</tr>
<tr>
<td></td>
<td>107289-03 is for the C compiler</td>
</tr>
<tr>
<td></td>
<td>107357-05 is for “BE” “build environment”</td>
</tr>
<tr>
<td></td>
<td>107311-09 is for the C++ compiler</td>
</tr>
<tr>
<td>Java 2 Software</td>
<td>Java 2 SDK 1.2.2_05 for Solaris.</td>
</tr>
<tr>
<td>Development Kit (SDK)</td>
<td></td>
</tr>
<tr>
<td>(SDK) for the</td>
<td></td>
</tr>
<tr>
<td>development</td>
<td></td>
</tr>
<tr>
<td>environment</td>
<td></td>
</tr>
</tbody>
</table>
### Table 6-16 Solaris SPARC 2.6 and Solaris SPARC 7 Software Requirements

<table>
<thead>
<tr>
<th>Software Requirements</th>
<th>Solaris SPARC 2.6 and Solaris SPARC 7 (32-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java 2 JRE for the run-time environment</td>
<td>JRE 1.2.2.05. This is needed for the BEA WebLogic EJB Deployer.</td>
</tr>
<tr>
<td>Non-BEA CORBA Java client</td>
<td>Java 2 SDK 1.2 IDL ORB (run time).</td>
</tr>
<tr>
<td>Internet browsers for the BEA Administration Console</td>
<td>Netscape 4.61.</td>
</tr>
<tr>
<td></td>
<td>Netscape 4.7.</td>
</tr>
<tr>
<td>Database for CORBA C++ applications</td>
<td>Oracle 7.3.4 on Solaris 2.6.</td>
</tr>
<tr>
<td></td>
<td>Oracle 8.1.5, also known as Oracle 8i, on Solaris 7.</td>
</tr>
<tr>
<td>Database for CORBA Java and J2EE applications</td>
<td>Oracle 7.3.4 (Type 2) on Solaris 2.6; no XA.</td>
</tr>
<tr>
<td></td>
<td>Oracle 8.1.5, also known as Oracle 8i, supports XA, on Solaris 7.</td>
</tr>
<tr>
<td>JDBC drivers</td>
<td>jdbcKona/Oracle 7.3.4 (Type 2), no XA.</td>
</tr>
<tr>
<td></td>
<td>WebLogic Enterprise JDBC/XA for Oracle 8.1.5, also known as Oracle 8i (Type 2); supports XA.</td>
</tr>
<tr>
<td>SSL certificate authorities</td>
<td>Verisign.</td>
</tr>
<tr>
<td></td>
<td>Netscape.</td>
</tr>
<tr>
<td>LDAP directory server</td>
<td>Netscape Enterprise Server.</td>
</tr>
</tbody>
</table>

### Additional Notes

- When using Oracle, Programmer/2000 Pro*C/C++ version 2.2.3.0.0 is required to build the WebLogic Enterprise University samples.
- EJB and RMI do not require C or C++.
- JDBC/XA users need a C compiler and linker to run the `buildtms` and `buildXAJS` commands.
Java IDL users need a C preprocessor. On UNIX systems, the C preprocessor comes with the system.

JNI users need a C or C++ compiler and linker.

BEA Tuxedo users need a C or C++ compiler or linker.

CORBA C++ users need a C++ compiler and linker.

For the BEA Administration Console, the table indicates browser support if the GUI is configured for 40-bit, 56-bit, or 128-bit encryption. If the GUI is configured for zero-bit encryption, Netscape 4.61 or later is supported.

When the optional WebLogic Enterprise Encryption Package software has been installed:

- The WebLogic Enterprise 5.1 ISL/ISH supports SSL 3.0 for IIOP connections.
- The WebLogic Enterprise 5.1 CORBA C++, CORBA Java, and RMI clients support SSL 3.0, and SSL connectivity between these clients and the WebLogic Enterprise ISL/ISH has been certified.
- Secure Sockets Layer (SSL) is not supported for Tuxedo client connections to the WebLogic Enterprise (Tuxedo) WSL/WSH. Tuxedo Link Level Encryption is available for encryption of these connections and encryption of connections between machines and domains.
- Although WebLogic Server supports SSL connections from its clients, the WebLogic Enterprise Connectivity feature in WLS 4.5.1 does not support SSL in the connection pools between WLS and the WebLogic Enterprise ISL/ISH. However, SSL connectivity is supported between the J-Engine and the T-Engine in WebLogic Enterprise 5.1.

To support certificate-based authentication when using SSL, WebLogic Enterprise provides an LDAP-based certificate retrieval mechanism. This has been certified for use with the LDAP Directory server included with Netscape Enterprise Server.

Network Requirements

TCP/IP using the TLI network interface.
Disk Space Requirements

The disk space requirements for installation on Solaris systems depends on which components you select during the installation, and whether other WebLogic Enterprise 5.1 components have been installed previously. Use the estimates in Table 6-17 as guidelines. These requirements are approximate and have been rounded up to the nearest megabyte (MB).

<table>
<thead>
<tr>
<th>Components</th>
<th>Solaris 2.6 and Solaris 7 (32-bit)</th>
</tr>
</thead>
<tbody>
<tr>
<td>All WebLogic Enterprise servers, all WebLogic Enterprise clients, and the BEA Administration Console server software</td>
<td>87 MB</td>
</tr>
<tr>
<td>Servers only</td>
<td>All servers: 69 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo server only: 22 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ server only: 43 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA Java server only: 68 MB</td>
</tr>
<tr>
<td></td>
<td>J2EE server only: 69 MB</td>
</tr>
<tr>
<td>Clients only</td>
<td>All clients: 29 MB</td>
</tr>
<tr>
<td></td>
<td>Tuxedo client only: 8 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA C++ client only: 20 MB</td>
</tr>
<tr>
<td></td>
<td>RMI/EJB client only: 6 MB</td>
</tr>
<tr>
<td></td>
<td>CORBA Java client only: 23 MB</td>
</tr>
<tr>
<td>Administration</td>
<td>BEA Administration Console: 12 MB</td>
</tr>
<tr>
<td>Encryption Packages, 56-bit or 128-bit</td>
<td>2 MB for LLE only on Tuxedo server or client system</td>
</tr>
<tr>
<td></td>
<td>4 MB for LLE and SSL</td>
</tr>
</tbody>
</table>

Installation Guide 6-43
Mounting and Unmounting the CD

The Solaris Volume Management software automatically mounts CDs on /cdrom/cdrom0/s0.

It is not necessary to unmount CDs on Solaris systems. However, it is necessary to issue a command to open the CD reader.

To open the CD reader, cd to root and enter eject.

Tuning Parameters

You probably need to reconfigure the Solaris kernel before running BEA WebLogic Enterprise software, because the default values of some tuning parameters are too low.

To adjust the tuning parameters, proceed as follows:

1. Determine whether the current values are adequate.
   For instructions about determining whether the current tuning parameter values are adequate, refer to “Verifying IPC Requirements” on page 4-17.

2. Reset the tuning parameters as necessary.
   Information regarding kernel configuration is provided in the Solaris systune(1M) man page.

Table 6-18 shows the default settings for the parameters and the settings used for the University sample applications. Use these settings as a starting point; however, your applications may require different settings.

The parameters currently set on your system are located in /etc/<systemname>.

Table 6-18 University Sample Applications Default Settings

<table>
<thead>
<tr>
<th>Solaris Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
<th>Setting for University Sample Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>shmsys:shminfo_shmmax</td>
<td>SHMMAX</td>
<td>131072</td>
<td>67108864</td>
</tr>
<tr>
<td>shmsys:shminfo_shmseg</td>
<td>SHMSEG</td>
<td>6</td>
<td>100</td>
</tr>
</tbody>
</table>
Table 6-18 University Sample Applications Default Settings (Continued)

<table>
<thead>
<tr>
<th>Solaris Name</th>
<th>Traditional Name</th>
<th>Default Setting</th>
<th>Setting for University Sample Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>shmsys:shminfo_shmmni</td>
<td>SHMMNI</td>
<td>100</td>
<td>300</td>
</tr>
<tr>
<td>semsys:seminfo_semmns</td>
<td>SEMMNS</td>
<td>60</td>
<td>5048</td>
</tr>
<tr>
<td>semsys:seminfo_semmni</td>
<td>SEMMNI</td>
<td>10</td>
<td>5029</td>
</tr>
<tr>
<td>semsys:seminfo_semmssl</td>
<td>SEMMSL</td>
<td>25</td>
<td>2000</td>
</tr>
<tr>
<td>semsys:seminfo_semmap</td>
<td>SEMMAP</td>
<td>10</td>
<td>5024</td>
</tr>
<tr>
<td>semsys:seminfo_semmnu</td>
<td>SEMMNU</td>
<td>30</td>
<td>1024</td>
</tr>
<tr>
<td>semsys:seminfo_semmume</td>
<td>SEMUME</td>
<td>10</td>
<td>128</td>
</tr>
<tr>
<td>msgsys:msginfo_msgmni</td>
<td>MSGMNI</td>
<td>50</td>
<td>1024</td>
</tr>
<tr>
<td>msgsys:msginfo_msgmap</td>
<td>MSGMAP</td>
<td>100</td>
<td>2048</td>
</tr>
<tr>
<td>msgsys:msginfo_msgmax</td>
<td>MSGMAX</td>
<td>2048</td>
<td>65535</td>
</tr>
<tr>
<td>msgsys:msginfo_msgmnb</td>
<td>MSGMNB</td>
<td>4096</td>
<td>65535</td>
</tr>
<tr>
<td>msgsys:msginfomsgsseg</td>
<td>MSGSSZ</td>
<td>8</td>
<td>256</td>
</tr>
<tr>
<td>msgsys:msginfo_msgseg</td>
<td>MSGSEG</td>
<td>1024</td>
<td>8192</td>
</tr>
<tr>
<td>maxusers</td>
<td>maxusers</td>
<td>32</td>
<td>200</td>
</tr>
<tr>
<td>max_nprocs</td>
<td>NPROC</td>
<td>10+16*maxusers</td>
<td>(MAXUSERS*3)+64</td>
</tr>
<tr>
<td>maxuprc</td>
<td>MAXUP</td>
<td>max_nprocs-5</td>
<td>(NPROC * 9) / 10</td>
</tr>
<tr>
<td>semsys:siminfo</td>
<td>semusz</td>
<td>1024</td>
<td></td>
</tr>
<tr>
<td>semsys:siminfo</td>
<td>semvmx</td>
<td>32767</td>
<td></td>
</tr>
<tr>
<td>semsys:siminfo</td>
<td>semaem</td>
<td>16384</td>
<td></td>
</tr>
</tbody>
</table>
Part III J-Engine Installation

Chapter 7. Overview of Installing WebLogic Server 5.1
Chapter 8. Installing WebLogic Server Using the InstallShield Distribution (Windows NT)
Chapter 9. Installing WebLogic Server from a Zip Archive (UNIX, Windows NT)
Chapter 10. Setting Up and Starting WebLogic Server 5.1
Chapter 11. Installing WebLogic jDriver for Oracle

BEA WebLogic Enterprise Installation Guide
CHAPTER

7 Overview of Installing WebLogic Server 5.1

This chapter 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


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 WebLogic Server Using the InstallShield Distribution (Windows NT)" on page 8-1 for more information.

- **Oracle users only:** The WebLogic jDriver for Oracle (formerly called jdbcKona/Oracle) is now available in several versions. The one you use 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 11-6.

- **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 the BEA WebLogic Enterprise box, it includes a J-Engine CD that contains installation packages for BEA WebLogic Server (including WebLogic Enterprise Connectivity) and BEA Jolt.

## Hardware Requirements

- 64 MB of RAM
- At least 80 MB of free hard disk space
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 10-24.

- **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. Rerun the WebLogic utility `ejbc` on your Enterprise Java Beans.

2. Rerun 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 WebLogic Server Using the InstallShield Distribution (Windows NT)” on page 8-1.
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 WebLogic Server from a Zip Archive (UNIX, Windows NT)” on page 9-1 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. Locate the weblogic510.exe file on your J-Engine 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.
Next Steps

**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 10-12 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 10-15.

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 11-1 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 10-1.
CHAPTER

9 Installing WebLogic Server 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” on page 10-15 will not be installed.
Upgrading from a Previous Release

- 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 WebLogic Server Using the InstallShield Distribution (Windows NT)” on page 8-1.

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. Locate the weblogic510.zip file on your J-Engine CD. The weblogic.zip file is located in the \wls directory which is 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,

   ```bash
   $ 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).
     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.
Installing from the Zip Archive

- 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” on page 10-9.

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.

<table>
<thead>
<tr>
<th>Platform</th>
<th>Directory containing shared library</th>
</tr>
</thead>
<tbody>
<tr>
<td>IBM AIX</td>
<td>weblogic/lib/aix</td>
</tr>
<tr>
<td>HP-UX 11</td>
<td>weblogic/lib/hpux11</td>
</tr>
<tr>
<td>SGI IRIX</td>
<td>weblogic/lib/irix</td>
</tr>
<tr>
<td>Red Hat Linux</td>
<td>weblogic/lib/linux</td>
</tr>
</tbody>
</table>

Installation Guide 9-4
Installing WebLogic Server from a Zip Archive (UNIX, Windows NT)

<table>
<thead>
<tr>
<th>Platform</th>
<th>Directory containing shared library</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sun Solaris</td>
<td>weblogic/lib/solaris</td>
</tr>
<tr>
<td>Compaq Tru64 UNIX</td>
<td>weblogic/tru64unix</td>
</tr>
</tbody>
</table>

(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 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 11-1 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 10-1.
Chapter 10

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 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” on page 10-14. The InstallShield distribution also allows Windows users to use “Windows Convenience Programs” on page 10-15 to start WebLogic Server.

Additional ways of starting WebLogic Server, including “Starting WebLogic Server from the Command Line” on page 10-18, and “Starting WebLogic Server Using Scripts” on page 10-24 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, at http://www.weblogic.com/docs51/cluster/index.html, in the WebLogic Server documentation.
Tasks to Set Up and Start WebLogic Server

1. “Installing a JDK” on page 10-2
2. “Setting the Environment” on page 10-4
   - “Setting the System PATH” on page 10-4
   - “Setting the Classpath” on page 10-4
   - “Setting up the Java Security Manager for Java 2” on page 10-9
   - “Installing a WebLogic License” on page 10-11
3. Starting WebLogic Server
   - “Important Note Regarding WebLogic RMI over IIOP” on page 10-18
   - “Starting WebLogic Server from the Command Line” on page 10-18
   - “Starting WebLogic Server from the WebLogic Console” on page 10-22
   - “Starting WebLogic Server Using Scripts” on page 10-24
   - “Starting WebLogic Server on Windows NT” on page 10-14
     - “Start Menu” on page 10-14
     - “NT Service” on page 10-14
     - “Windows Convenience Programs” on page 10-15
     - “Starting WebLogic Server from the Command Line” on page 10-18
4. “Next Steps” on page 10-24
   - “Install JDBC Drivers for Use with WebLogic Server” on page 10-24
   - “Other Documentation” on page 10-25

Installing 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 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 Versus Green Threads**

Most JSDKs 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 JSDKs. Consult the documentation for you 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 JSDKs 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 11-1 for more information. On Windows NT, you would use the following command to set your path in a command window:

```bash
$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 10-7.

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/APIejb/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” on page 10-9.)
- `/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;
```
Setting the Environment

\begin{verbatim}
c:/weblogic/lib/weblogicaux.jar;
c:/weblogic/myserver/serverclasses weblogic.Server
\end{verbatim}

Where \texttt{c:/weblogic} is the path to the directory where you installed WebLogic Server and \texttt{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:

\begin{verbatim}
-Dweblogic.classloader.verbose=true.
\end{verbatim}

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 \texttt{jview} is different from the \texttt{java} command. To run WebLogic Server under \texttt{jview}, make the following substitutions:

- \texttt{jview} for \texttt{java}
- \texttt{/d:} for the \texttt{-D} option (to set properties)

Note that \texttt{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 \texttt{weblogic.class.path} property). Since the class loader is not used, you will not be able to use the Hot Deploy feature for deploying EJ Bs 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 \texttt{weblogic.class.path} property to load classes:

\begin{verbatim}
/d:weblogic.system.disableWeblogicClassPath=true
\end{verbatim}

To run WebLogic Server under jview, do not specify the \texttt{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 \texttt{CLASSPATH}, along with the classes which normally belong in the Java system classpath.
Setting Up and Starting WebLogic Server 5.1

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 at http://www.weblogic.com/docs51/classdocs/API_com.html for more information.

The WebLogic Frequently Asked Questions also has a section on 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 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 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” on page 10-12.

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>
Setting the Environment

<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 (see http://www.weblogic.com/docs51/techstart/utils.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 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 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 10-15 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 10-15.

For additional information, see Using WebLogic Server as an NT 4.0 service at [http://www.weblogic.com/docs51/admindocs/ntservice.html](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,
   ```bash
   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:
   ```bash
   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](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:

```bash
$ 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.
Starting WebLogic Server on Windows NT

- help
  Prints a list of optional arguments for the \texttt{wlconfig.exe} command.

- msSizeInMb
  Sets the initial Java heap size (in megabytes).
  See WebLogic Server Performance Tuning Guide at \url{http://www.weblogic.com/docs51/admindocs/tuning.html#jvm}
  \texttt{execution} for information on setting heap size.

- mxSizeInMb
  Sets the maximum Java heap size (in megabytes).
  See WebLogic Server Performance Tuning Guide at \url{http://www.weblogic.com/docs51/admindocs/tuning.html#jvm}
  \texttt{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, \texttt{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 \texttt{wlserver.exe}, the Start menu, and when starting as an NT service.

\texttt{wlconsole.exe}
Runs the WebLogic Console, an administrative tool.

\texttt{wlserver.exe}
Runs WebLogic Server with the defaults defined by running \texttt{wlconfig.exe} (see above).

\texttt{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 10-24.

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
Starting WebLogic Server from the Command Line

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

- Set your `-classpath` option and `weblogic.classpath` as shown in “Setting
your WebLogic Classpath” on page 10-6. These are the minimum requirements
for classpath:
  - `-classpath c:/java/lib/classes.zip;c:/weblogic/classes;`  
    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/umserver/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 10-9 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:

```bash
-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**

```bash
$ 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**

```bash
$ 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
```

10 Setting Up and Starting WebLogic Server 5.1

10-20 Installation Guide
Starting WebLogic Server from the Command Line

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 10-7.

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 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 10-24.

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 `JDK_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:

  ```bash
  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 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 11-1. 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.


Informix


Microsoft SQL Server


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

- WebLogic Developer Center (http://www.weblogic.com/docs51/resources.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
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 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 (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 11-6.

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 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
Installing WebLogic jDriver for Oracle

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 at [http://www.weblogic.com/docs51/admindocs/classpath.html](http://www.weblogic.com/docs51/admindocs/classpath.html). If you are 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 at [http://www.weblogic.com/docs51/techsupport/appletsecurity.html](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” on page 11-5.

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” on page 11-7, 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

<table>
<thead>
<tr>
<th>Oracle Client version</th>
<th>Oracle API version</th>
<th>Directory</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.3.4</td>
<td>7</td>
<td>oci734_7</td>
<td>Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.0.5</td>
<td>7</td>
<td>oci805_7</td>
<td>Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.0.5</td>
<td>8</td>
<td>oci805_8</td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.1.5</td>
<td>8</td>
<td>oci815_8</td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
</tbody>
</table>

### 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

<table>
<thead>
<tr>
<th>Oracle Client version</th>
<th>Oracle API version</th>
<th>Directory</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.3.4</td>
<td>7</td>
<td>oci734_7</td>
<td>Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.0.5</td>
<td>7</td>
<td>oci805_7</td>
<td>Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.0.5</td>
<td>8</td>
<td>oci805_8</td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.0.6</td>
<td>8</td>
<td>oci806_8</td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.1.5</td>
<td>8</td>
<td>oci815_8</td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
</tbody>
</table>
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 at [http://www.weblogic.com/docs51/platforms/index.html#jdbc](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)

<table>
<thead>
<tr>
<th>Oracle Client version</th>
<th>Oracle API version</th>
<th>Directory</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>8.0.4</td>
<td>7</td>
<td><code>oci804_7</code></td>
<td>Provided for backwards compatibility. No Oracle 8 or JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.0.5</td>
<td>8</td>
<td><code>oci805_8</code></td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
<tr>
<td>8.1.5</td>
<td>8</td>
<td><code>OCI815_8</code></td>
<td>Allows access to Oracle 8 and JDBC 2.0 features.</td>
</tr>
</tbody>
</table>

**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 at [http://www.weblogic.com/docs51/platforms/index.html#jdbc](http://www.weblogic.com/docs51/platforms/index.html#jdbc).
Installing WebLogic jDriver for Oracle

- 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 \texttt{LD_LIBRARYN32_PATH}.

Where \texttt{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 at \url{http://www.weblogic.com/docs51/platforms/index.html#jdbc}.

Add the directory weblogic/lib/reliantunix to your \texttt{LD_LIBRARY_PATH}.

Where \texttt{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 at \url{http://www.weblogic.com/docs51/platforms/index.html#jdbc}.

Add the directory weblogic/lib/tru64unix to your \texttt{LD_LIBRARY_PATH}.

Where \texttt{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, 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.
11 Installing WebLogic jDriver for Oracle

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 11-6). 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.

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


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_jdbc3.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
11  *Installing WebLogic jDriver for Oracle*

**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.

Part IV Encryption Package Installation

Chapter 12. WebLogic Enterprise Encryption Package Installation on Windows Systems
Chapter 13. WebLogic Enterprise Encryption Package Installation on UNIX Systems
This chapter explains how to install the following optional BEA WebLogic Enterprise Encryption Package software products on Windows NT systems:

- WebLogic Enterprise 5.1 56-bit Encryption Package
- WebLogic Enterprise 5.1 128-bit Encryption Package

This topic includes the following sections:

- Before You Install
- Platforms Supported
- Installing the WebLogic Enterprise 5.1 Encryption Package on Microsoft Windows Systems
- Removing (Uninstalling) the WebLogic Enterprise Encryption Package from Your System
WebLogic Enterprise Encryption Package Installation on Windows Systems

Notes: The Encryption Package software described in this chapter is distinct and separate from the SSL software that you install on J-Engine server systems, as described in Part II of this document.

When installing the Encryption Package software (formerly called Security Service) on a Tuxedo-only system, only the encryption libraries are updated. None of the SSL files are installed, nor are the SSL plug-ins registered. When installing on any other system (CORBA C++, CORBA Java, etc.), the encryption libraries are updated, the information about SSL is prompted for, the SSL files are installed, and the plug-ins are registered.

Also note that the Encryption software cannot be installed on a RMI/EJB client-only system.

For information about installing WebLogic Enterprise Encryption Package software products on a UNIX system, see Chapter 13, “WebLogic Enterprise Encryption Package Installation on UNIX Systems.”

The WebLogic Enterprise Encryption Package software products are packaged on a CD that is separate from the WebLogic Enterprise product box. A WebLogic Enterprise Encryption Package CD is distributed only if you purchased this software. This software provides 56-bit or 128-bit Secure Sockets Layer (SSL) and Link-Level Encryption (LLE) features for WebLogic Enterprise applications. Each level of encryption is packaged on a separate CD.

The installation screens are similar for both levels of security. In this chapter, the sample screens are from a WebLogic Enterprise128-bit Encryption Package product installation on Microsoft Windows NT. Installation screens that are identical to the WebLogic Enterprise T-Engine installation screens are described, but not shown, in this chapter.

Before You Install

This topic includes the following sections:

- Installing a WebLogic Enterprise 5.1 Encryption Package Product to Support WebLogic Enterprise Connectivity
- Confirming That the WebLogic Enterprise 5.1 Software Has Been Installed
Before You Install

- LDAP Information Required During the Installation
- Before Re-installation, Backup LDAP Files
- Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services
- Check That Your Account Has Administrator Privileges

Installing a WebLogic Enterprise 5.1 Encryption Package Product to Support WebLogic Enterprise Connectivity

If you are installing a WebLogic Enterprise 5.1 Encryption Package product to support WebLogic Enterprise Connectivity (WLEC), you need to take the following steps:

1. Confirm that the WebLogic Enterprise 5.1 J-Engine server components have been installed on your machine, as described in Part II of this document.

2. Install either the WebLogic Enterprise CORBA C++ or CORBA Java client software on your machine, as described in the section “Microsoft Windows 2000 or NT 4.0 Installation Procedure” on page 2-8.

3. Complete the remaining steps described in the current chapter.

Confirming That the WebLogic Enterprise 5.1 Software Has Been Installed

Before you can install a WebLogic Enterprise 5.1 Encryption Package software product, you must first install at least one WebLogic Enterprise 5.1 server component, or one of the following WebLogic Enterprise 5.1 client component options:

- All WebLogic Enterprise client components (recommended)
- BEA CORBA C++ client
- BEA CORBA Java client
If you are installing a WebLogic Enterprise 5.1 Encryption Package software product on a Windows 98 or Windows 95 client system, you must first install one of the WebLogic Enterprise 5.1 client component options shown in the previous list.

If the target system only has the Tuxedo server or client software installed from the WebLogic Enterprise 5.1 installation, the WebLogic Enterprise 5.1 Encryption Package installation procedure only installs the Link-Level Encryption (LLE) components. The Secure Sockets Layer (SSL) components are not installed on this type of target system.

LDAP Information Required During the Installation

During the 56-bit or 128-bit Encryption Package installation, the procedure prompts you for the required LDAP server information shown in the following list.

If you do not know the appropriate LDAP values for the prompts, contact the person in your organization or company who is responsible for defining the LDAP server tree. At most companies, this person is the Security Administrator or the Directory Services Administrator.

**Note:** After the installation, it is not possible to modify a file to adjust these values. The only way to change these values is to re-install the WebLogic Enterprise Encryption Package software and specify the updated values. Therefore, it is important that you understand the appropriate values for the requested information before you start the installation.

- The hostname of the LDAP server computer system.
- The port on the LDAP server computer system that is listening for requests.
- An appropriate base object in the LDAP server tree. The **base object** is the point in the LDAP tree at which you want users to start searching for certificates. By defining a specific location in the LDAP tree, you can narrow the scope of the search for certificates on the relevant portion of the LDAP server tree, and avoid longer-than-necessary searches through irrelevant portions of the LDAP server tree.

**Note:** These LDAP prompts are not presented if the target system only has the Tuxedo server or client software installed (from WebLogic Enterprise 5.1). In this case, only the WebLogic Enterprise Encryption Package’s Link-Level
Encryption (LLE) components are installed on the target system. During the WebLogic Enterprise Encryption Package installation procedure, the Secure Sockets Layer (SSL) components are not installed on this type of target system.

**Before Re-installation, Backup LDAP Files**

If you are re-installing the 56-bit or 128-bit WebLogic Enterprise Encryption Package software on a system, the installation procedure will overwrite the LDAP filter file if you selected its default name and location. By default, the LDAP filter file is installed in `%TUXDIR%\udataobj\security\bea_ldap_filter.dat`, where `TUXDIR` is the directory in which you installed the WebLogic Enterprise software. This file is used to define search filters that can further refine the scope of searches in the LDAP server tree.

On re-installation, the Encryption Package installation procedure will also overwrite the LDAP peer validation rule file,

`%TUXDIR%\udataobj\security\peer_val.rul`.

Before you re-install the Encryption Package software, temporarily rename these files if you do not want the installation procedure to overwrite them. After the installation procedure, rename the files back to their original names and locations.

**Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services**

Before beginning the installation, make sure no BEA Tuxedo or WebLogic Enterprise client or server applications are running. For information about the `tmshutdown` command, see *Starting and Shutting Down Applications* in the Administration section of the WebLogic Enterprise online documentation.
Check That Your Account Has Administrator Privileges

You need administrator privileges to perform the installation. If you attempt to install the WebLogic Enterprise 5.1 Encryption Package software without administrator privileges, the following error message will be displayed:

Cannot Install Tuxedo IPC Helper Service.

Platforms Supported

The Microsoft Windows platforms listed in Table 12-1 are supported.

Table 12-1  Supported Microsoft Windows Platforms

<table>
<thead>
<tr>
<th>Operating System</th>
<th>Release/Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft Windows 2000</td>
<td></td>
</tr>
<tr>
<td>Microsoft Windows NT</td>
<td>4.0 (Intel) Service Pack 5 (SP5)</td>
</tr>
<tr>
<td>Microsoft Windows 95</td>
<td>Service Pack 1</td>
</tr>
<tr>
<td>Microsoft Windows 98</td>
<td></td>
</tr>
</tbody>
</table>

For the hardware and software requirements for NT, see Chapter 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
Installing the WebLogic Enterprise 5.1 Encryption Package on Microsoft Windows Systems

This section describes how to install the WebLogic Enterprise 56-bit or 128-bit Encryption Package software products on Microsoft Windows systems. The sample screens show the installation of the 128-bit Encryption Package software on a Windows NT system.

Microsoft Windows Installation Procedure

It will take approximately 5 minutes to install the software.

To install the WebLogic Enterprise 56-bit or 128-bit Encryption Package software on a Microsoft Windows system, perform the following steps:

1. Insert the WebLogic Enterprise Encryption Package software CD in the CD player. The Windows autorun feature is used to automatically begin the installation. (To bypass the autorun feature, hold down the Shift key while inserting the CD in the CD player. You can then run the setup.exe file in the inwnt40 directory on the CD.)

2. The Setup screen is displayed, followed by the Welcome screen.
3. Click Next. The Software License Agreement screen is displayed. To accept the license agreement, click Yes.

4. The User Information screen is displayed. Enter your name and the name of your company and click Next.

5. If a WebLogic Enterprise or BEA Tuxedo application and related services are running, the Severe message screen is displayed:
If this happens, click OK, which will exit the installation without installing any portion of the software. Then use the tmshutdown command, if appropriate, to stop any running applications. Also stop any WebLogic Enterprise or BEA Tuxedo services, if appropriate. You can do this on Windows NT systems by clicking Start—>Settings—>Control Panel. In the Control Panel, open the Services panel. The Services screen is displayed.

Scroll to the entries for Tuxedo IPC Helper and TLiisten. For each entry, select it in the display box and click Stop to change its status from Started to Stopped. Then click Close. If you encountered the screens shown in this step, you will have to restart the installation program.

6. If you did not encounter the problem described in the preceding step, the installation program displays the Enter LDAP Server URL And Port Information screen.

**Note:** If you installed only BEA Tuxedo server or client software on this system, the following screen is not displayed.
This information will be stored locally as a registered SSL certificate lookup plug-in that WebLogic Enterprise client and server applications can use.

- In the Hostname input box, enter the LDAP server’s fully qualified node name and domain name.
- In the Port input box, enter the port on which the LDAP server is listening.

7. The Base Object screen is displayed.

**Note:** If you installed only BEA Tuxedo server or Tuxedo client software on this system, the following screen is not displayed.
8. The base object is the point in the LDAP tree at which you want users to start searching; in this case, to start searching for certificates. There are no strict rules about the syntax for this value. Enter the base object string exactly as it was specified in the LDAP server tree. Click Next. The Enter LDAP Filter File Information screen is displayed.

**Note:** If you installed only BEA Tuxedo server or client software on this system, the following screen is not displayed.
9. The default location for the LDAP filter file is
\wledir\udataobj\security\bea_ldap_filter.dat. This file is used to
define search filters that can further refine the scope of searches in the LDAP
server tree. (For more information, see Using Security in the WebLogic
Enterprise online documentation.) Click Next to accept this default; or, enter a
new value and click Next. The Setup Complete screen is displayed:
10. Click Finish to complete the installation.

11. Reboot your system. Your system restarts. If you attempt to run the WebLogic Enterprise software before you restart your system, the software may fail.
Removing (Uninstalling) the WebLogic Enterprise Encryption Package from Your System

To remove a WebLogic Enterprise Encryption Package product from your system, follow the steps in this section.

1. Log on to the system. If you are using a Microsoft Windows NT system, log on as the administrator or as a member of the Administrator group.

2. Make sure that no BEA Tuxedo or WebLogic Enterprise client or server applications are running. Use `tmshutdown` to shut down all WebLogic Enterprise applications.


4. The Confirm File Deletion screen is displayed. For example:

```
Confirm File Deletion

Are you sure you want to completely remove 'BEA 128-bit Encryption Package' and all of its components?

Yes  No
```

5. Click Yes to confirm the removal and to uninstall the WebLogic Enterprise Encryption Package product. The Uninstall screen is displayed. For example:
Removing (Uninstalling) the WebLogic Enterprise Encryption Package from Your Sys-

The WebLogic Enterprise Encryption Package product is removed from your system and from the Windows Registry.
CHAPTER 13

WebLogic Enterprise Encryption Package
Installation on UNIX Systems

This chapter explains how to install the following optional BEA WebLogic Enterprise Encryption Package software products on the supported UNIX systems:

- WebLogic Enterprise 5.1 56-bit Encryption Package
- WebLogic Enterprise 5.1 128-bit Encryption Package

This topic includes the following sections:

- Before You Install
- Platforms Supported
- Installing WebLogic Enterprise Encryption Package on UNIX Systems
- Removing (Uninstalling) the WebLogic Enterprise Encryption Package Software from Your System
Notes: The Encryption Package software described in this chapter is distinct and separate from the SSL software that you install on J-Engine server systems, as described in Part II of this document.

When installing the Encryption Package software (formerly called Security Service) on a Tuxedo-only system, only the encryption libraries are updated. None of the SSL files are installed, nor are the SSL plug-ins registered. When installing on any other system (CORBA C++, CORBA Java, etc.), the encryption libraries are updated, the information about SSL is prompted for, the SSL files are installed, and the plug-ins are registered.

Also note that the Encryption software cannot be installed on a RMI/EJB client-only system.

For information about installing WebLogic Enterprise Encryption Package software on a Microsoft Windows system, see Chapter 12, “WebLogic Enterprise Encryption Package Installation on Windows Systems.”

The WebLogic Enterprise Encryption Package software is packaged on a CD that is separate from the WebLogic Enterprise product box. A WebLogic Enterprise Encryption Package CD is distributed only if you purchased this software. This software provides 56-bit or 128-bit Secure Sockets Layer (SSL) and Link Level Encryption (LLE) features for WebLogic Enterprise applications. Each level of encryption is packaged on a separate CD.

The installation screens are similar for both levels of security. In this chapter, the sample screens are from a WebLogic Enterprise 5.1 56-bit Encryption Package installation on a Sun Solaris 2.6 system.

Before You Install

This topic includes the following sections:

- Installing a WebLogic Enterprise 5.1 Encryption Package Product to Support WebLogic Enterprise Connectivity
- Confirming That the WebLogic Enterprise 5.1 Software Has Been Installed
- Environment Variables
Before You Install

- LDAP Information Required During the Installation
- Before Re-installation, Back Up LDAP Files
- Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services
- Checking That Your Account Has Administrator Privileges

Installing a WebLogic Enterprise 5.1 Encryption Package Product to Support WebLogic Enterprise Connectivity

If you are installing a WebLogic Enterprise 5.1 Encryption Package product to support WebLogic Enterprise Connectivity (WLEC), you need to take the following steps:

1. Confirm that the WebLogic Enterprise 5.1 J-Engine server components have been installed on your machine, as described in Part II of this document.

2. Install either the WebLogic Enterprise CORBA C++ or CORBA Java client software on your machine, as described in the section “UNIX Installation Procedure” on page 3-4.

3. Complete the remaining steps described in the current chapter.

Confirming That the WebLogic Enterprise 5.1 Software Has Been Installed

Before you can install the WebLogic Enterprise 5.1 Encryption Package software, you must first install at least one WebLogic Enterprise 5.1 server component, or one of the following WebLogic Enterprise 5.1 client component options:

- All WebLogic Enterprise client components (recommended)
- BEA CORBA C++ client
- BEA CORBA Java client
Environment Variables

The environment variables discussed in the section “Setting Up Your Environment on UNIX Systems” on page 4-12 must be set prior to installing the WebLogic Enterprise Encryption Package software. The _TUXDIR_ and dynamic shared library path variables are critical to the success of this Encryption Package installation, because the SSL plug-in registration step depends on these variables.

LDAP Information Required During the Installation

During the installation of either the 56-bit or the 128-bit Encryption Package, the procedure will prompt you for the required LDAP server information shown in the following list.

If you do not know the appropriate LDAP values for the prompts, contact the person in your organization or company who is responsible for defining the LDAP server tree. At most companies, this person is the Security Administrator or Directory Services Administrator.

**Note:** After the installation, it is not possible to modify a file to adjust these values. The only way to change these values is to re-install the product. Therefore, it is important that you understand the appropriate values for the requested information before you start the installation.

- The hostname of the LDAP server computer system.
- The port on the LDAP server computer system that is listening for requests.
- An appropriate base object in the LDAP server tree. The **base object** is the point in the LDAP tree at which you want users to start searching for certificates. By defining a specific location in the LDAP tree, you can narrow the scope of the search for certificates on the relevant portion of the LDAP server tree, and avoid longer-than-necessary searches through irrelevant portions of the LDAP server tree.

**Note:** These LDAP prompts are not presented if the target system only has the Tuxedo server or client software installed (from WebLogic Enterprise 5.1). In this case, only the WebLogic Enterprise Encryption Package’s Link-Level Encryption (LLE) components are installed on the target system. During the
WebLogic Enterprise Encryption Package installation procedure, the Secure Sockets Layer (SSL) components are not installed on this type of target system.

Before Re-installation, Back Up LDAP Files

If you are re-installing either the WebLogic Enterprise 56-bit or 128-bit Encryption Package software on a system, the installation procedure will overwrite the LDAP filter file if you selected its default name and location. By default, the LDAP filter file is installed in \$TUXDIR/udataobj/security/bea_ldap_filter.dat, where TUXDIR is the directory in which you installed the WebLogic Enterprise software. The filter file is used to define search filters that can further refine the scope of searches in the LDAP server tree.

On re-installation, the Encryption Package installation procedure will also overwrite the LDAP peer validation rule file, \$TUXDIR/udataobj/security/peer_val.rul.

Before you re-install the Encryption Package software, temporarily rename these files if you do not want the installation procedure to overwrite them. After the installation procedure, rename the files back to their original names and locations.

Stopping WebLogic Enterprise or BEA Tuxedo Applications and Related Services

Before beginning the installation, ensure that no BEA Tuxedo or WLE client or server applications are running. For information about the tmshutdown command, see Starting and Shutting Down Applications in the Administration section of the WebLogic Enterprise online documentation.
Checking That Your Account Has Administrator Privileges

On most systems, you need superuser privileges to mount the software CD. The account that you log on to perform the installation must have administrative privileges.

Platforms Supported

The platforms listed in Table 13-1 are supported.

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Operating System</th>
<th>Release/Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compaq</td>
<td>Tru64 UNIX</td>
<td>4.0f on Alpha systems</td>
</tr>
<tr>
<td>HP</td>
<td>HP-UX</td>
<td>11.00 32-bit plus patches B.11.00.B0315</td>
</tr>
<tr>
<td>IBM</td>
<td>AIX</td>
<td>4.3.3</td>
</tr>
<tr>
<td>SCO</td>
<td>Unixware</td>
<td>7.1.1 (C++ only)</td>
</tr>
<tr>
<td>Sun Microsystems</td>
<td>Solaris</td>
<td>Solaris 2.6 and Solaris 7 (SPARC)</td>
</tr>
</tbody>
</table>

For the hardware and software requirements for these operating systems, see Appendix 6, “WebLogic Enterprise T-Engine Platform Data Sheets.”
Installing WebLogic Enterprise Encryption Package on UNIX Systems

This section describes how to install the WebLogic Enterprise 56-bit or 128-bit Encryption Package software on the supported UNIX systems. The sample screens show the installation of the 56-bit Encryption Package software on a Solaris 2.6 system.

UNIX Installation Procedure

It takes approximately 10 minutes to install the software.

To install the WebLogic Enterprise Encryption Package software on a UNIX operating system, perform the following steps:

1. Log on to the system with administrative privileges.

2. Insert the WebLogic Enterprise Encryption Package CD into the reader.

3. Mount the CD as a file system. For platform-specific instructions on how to do this, see Appendix 6, “WebLogic Enterprise T-Engine Platform Data Sheets.” On most systems you need superuser privileges to perform the mount.

   Note: If your system does not have a directly connected CD reader, you can mount the CD on a remote system, share (export) the CD file system, and then mount the remote file system. For detailed instructions for each platform, see Appendix 6, “WebLogic Enterprise T-Engine Platform Data Sheets.” Alternatively, you can mount the CD on a remote system, copy the contents of the CD directory for your platform to the system in which you plan to install the WebLogic Enterprise software, and continue with the remainder of the installation procedure.

4. Use the cd command to change your working directory to the root of the WebLogic Enterprise Encryption Package software CD.

5. Run the ls command in the root directory to check the CD’s contents. If all the files are in lowercase characters, begin the installation by entering:
sh install.sh.

If all the files are in uppercase characters, begin the installation by entering:

sh INSTALL.SH

6. The installation procedure lists the available platform choices. Enter the number that corresponds to the installation’s target platform.

7. The remaining prompts in this chapter show a sample Encryption Package application on a Solaris 2.6 system. For example, a confirmation prompt is displayed:

** You have chosen to install software for **
BEA WebLogic Enterprise Release 5.1
This directory contains the BEA WLE Installation Software for Sun Solaris v2.6 on Sun SPARC.
Is this correct? [y,n,q]:

Enter y to proceed; or enter n to redisplay the platform menu; or enter q to quit the installation.

8. If you entered y, a component menu is displayed:

To terminate the installation at any time, press the interrupt key, typically <del>, <break>, or <ctrl+c>.
The following components are available:

1 security  BEA Encryption Package 56

Select the one you wish to install [?,??,q]:

Enter the number 1 to select the Encryption Package; or enter a single question mark (?) to display a brief help message; or enter two question marks (??) to redisplay the menu; or enter q to quit the installation.

9. If you entered the number 1 or pressed the Enter key, a packages menu is displayed:

The following packages are available:

1 sec56  BEA Encryption Package 56 For WLE

Select the package(s) you wish to install (or ‘all’ to install all packages) (default: all) [?,?,??,q]:

13-8  Installation Guide
Installing WebLogic Enterprise Encryption Package on UNIX Systems

Enter the number 1 or the word all to install the Encryption Package for WebLogic Enterprise; or enter a single question mark (?) to display a brief help message; or enter two question marks (??) to redisplay the menu; or enter q to quit the installation.

10. If you entered the number 1 or the word all, the following messages are displayed:

BEA Encryption Package 56 For WLE
(sparc) Release 5.1
Copyright (c) 1999 BEA Systems, Inc.
All Rights Reserved.
BEA and WebLogic are trademarks of BEA Systems, Inc.
SSLplus is a trademark of Certicom Corporation, 1999-2000.

WebLogic Enterprise must be installed prior to installing the Encryption Package

11. The installation program checks for existing BEA software and prompts you for the WebLogic Enterprise base directory:

Location of existing BEA software installation (default: /usr/local/wledir) [?,q]:

Press the Enter key if the default value shown matches the base directory location of the WebLogic Enterprise software; or enter the correct path to the WebLogic Enterprise base directory.

12. If the installation program finds the WebLogic Enterprise software in the location specified, the installation continues. A confirmation message is displayed, and then the installation program checks for sufficient disk space. For example:

Using /usr/local/wledir as the base directory
Determining if sufficient space is available ... 5818 blocks are required
1032768 blocks are available to /usr/local/wledir

13. If sufficient space is found, the installation program starts moving files to the target system and displays messages.

Note: In the following displays and steps, all the SSL-related messages and prompts starting with "Unloading...SECSSL.Z" through "Registering SSL plug-in...finished" (in step 20) are not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation. In this case, the Encryption
Package installation procedure installs the LLE software, but not the SSL software.

Moving /usr/local/wledir/lib/libgp.so.65 to /usr/local/wledir/lib/libgp.so.65.0
Moving /usr/local/wledir/lib/libgp.a to /usr/local/wledir/lib/libgp.a.0

Unloading /usr/local/wledir/spsol26/security/sec56/SEC56.Z ...
lib/libgp.so.65
lib/libgp.a
2750 blocks
... finished

Unloading /usr/local/wledir/spsol26/security/sec56/SECSSL.Z ...
lib/liborbssl.so.65
lib/libjsec.so
lib/libsecssl.so.65
lib/libwlesec.so.65
locale/C/IJSSLN.text
locale/C/IJSSLN_CAT
udataobj/security/bea_ldap_filter.dat
udataobj/security/certs/peer_val.rul
udataobj/security/certs/revoked.crl
udataobj/security/certs/trust_ca.cer
2970 blocks
... finished

14. Enter information about the LDAP server. This information will be stored locally as a registered SSL certificate lookup plug-in that WebLogic Enterprise client and server applications can use. The following prompt is displayed:

Enter fully qualified hostname for URL of the LDAP server system. [?,q]:

Enter the LDAP server’s fully qualified node name and domain, such as myhost.mydomain.com.

**Note:** This prompt is not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation.

15. Enter the port number on which the LDAP server will be listening for certificate requests:

Enter a port number for the URL of the LDAP server system. [?,q]:

For example, enter 389 if that is the correct port number. If you are not sure, check the value with the system administrator of the LDAP server.
16. The installation program displays a confirmation message:

   Using ‘myhost.mydomain.com:389’ as the URL of the LDAP server/port

Enter a base object for searches in the LDAP server. The base object is the point in the LDAP tree at which you want users to start searching (in this case, to start searching for certificates). There are no strict rules about the syntax for this value. Enter the base object string exactly as it was specified in the LDAP server tree.

Enter a base object for search in LDAP server. [?,q]:

   o=mydomain.com

For example, you could enter a value such as o=mydomain.com, or a value such as o=trixie@trixieweb.com.

Note: This prompt is not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation.

17. The installation program displays a confirmation message for the value you entered:

   Using ‘o=mydomain.com’ as the base object

Note: This message is not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation.

18. Enter the location for the LDAP filter file. This file is used to define search filters that can further refine the scope of searches in the LDAP server tree. For more information, see Using Security in the WebLogic Enterprise online documentation.

   Location and name of LDAP filter file. (default: /usr/local/wledir/udataobj/security/bea_ldap_filter.dat) [?,q]:

   The file’s default location is shown. Press the Enter key to accept this default; or enter a new value and click Enter.

Note: This prompt is not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation.

19. The installation program displays a confirmation message for the value you entered:
Using
‘file:///usr/local/wledir/udataobj/security/bea_ldap_filter.dat
’ as the location and name of LDAP filter file

Note: This message is not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation.

20. The installation program then completes the installation and displays confirmation messages:

Registering SSL plug-in...
... finished

Note: The Registering SSL... message is not displayed if the system only has Tuxedo server or client software from a WebLogic Enterprise 5.1 software installation.

Changing file permissions...
... finished

Installation of BEA Encryption Package 56 For WLE was successful

Please don’t forget to fill out and send in your registration card

Removing (Uninstalling) the WebLogic Enterprise Encryption Package Software from Your System

To remove either the WebLogic Enterprise 56-bit or 128-bit Encryption Package software from your UNIX system, you must:

1. Uninstall the WebLogic Enterprise 5.1 software, as explained in the section “Removing (Uninstalling) the WebLogic Enterprise Software from Your System” on page 3-19.
   This also removes the WebLogic Enterprise Encryption Package software, if present.

2. Then re-install the WebLogic Enterprise 5.1 software.
Part V  BEA Jolt
Installation

Chapter 14. Preparing to Install BEA Jolt
Chapter 15. Installing BEA Jolt
Chapter 16. Configuring the BEA Jolt System
Chapter 17. Post Installation Considerations for BEA Jolt

BEA WebLogic Enterprise Installation Guide
This chapter provides information on the preparations you need to make prior to installing BEA Jolt, hereafter designated as Jolt. The following topics are discussed in this chapter:

- What’s New in Jolt
- What’s Changed in Jolt
- System Requirements
- Release Migration/Interoperability
- Jolt ASP Connectivity for BEA Tuxedo
- Preinstallation Checklist

What’s New in Jolt

The following features are new to Jolt:

- **Compression**—Allows application data that is sent between a Jolt client and a Jolt Server (JSH) to be compressed during transmission over the network at a very low cost (few CPU cycles). Compression is only supported for clients based on JDK 1.1.7 or 1.2.
Diffie-Hellman (DH) Key Exchange—Replaces the DES key exchange mechanism. This allows Jolt to be classified as a true 128 bit secure environment (available to U. S. customers only).

Two Session Keys per Jolt Session—Jolt uses two session keys for each user session. One key is used for encrypting and decrypting messages sent from the client to the server, the other for messages sent from the server to the client.

Support for WebLogic Enterprise 5.1—WebLogic Enterprise 5.1 provides advanced Java development services. It allows you to build distributed, mission-critical, CORBA-compliant applications in Java and C++.

JRLY as an NT Service—The Jolt Relay (JRLY) in Jolt on the NT platform is provided as an NT service. As an NT service, this process can be started and stopped via the NT Service Control Manager (SCM).

JRLY Connection to JRAD—The Jolt Relay (JRLY) in Jolt allows you to specify a list of IP addresses for JRADs. On startup, the JRLY tries to connect to each JRAD on the list, searching sequentially from the beginning. The first JRAD to respond successfully is used.

Multiple JSL Addresses for JoltSessionAttributes—Jolt allows you to specify a list of IP port addresses from which the Jolt client randomly selects JSLs until a connection is established.

ASP Connectivity for BEA Tuxedo (formerly Jolt WAS for IIS)—Works with existing Microsoft Internet Information Server (IIS) to provide a gateway for HTML clients into a BEA Tuxedo application environment. Interactions between the Web server and Jolt classes are through VBScript and VB inside ASPs.

JSE Connectivity for BEA Tuxedo (formerly Jolt WAS for Servlets)—Simplifies handling of HTTP requests to the BEA Tuxedo application environment.

Internationalization—Allows all informational and error messages that are generated by the Jolt client to be localized.

Javadoc Format—The HTML and PDF versions of the Jolt class library are now in Javadoc format.

Y2K Compliance—Any dates that are used or displayed by Jolt are in Y2K-compliant format. Specifically, the Jolt license file allows expiration dates beyond December 31, 1999 and all messages and timestamps that Jolt generates use a Y2K-compliant representation.
What’s Changed in Jolt

- **BEA Tuxedo/WebLogic Enterprise Version** — Jolt 1.2 for WebLogic Enterprise 5.1 only works with WebLogic Enterprise 5.1.

- **Client/Server Interoperability** — Jolt provides some client/server interoperability. While you must upgrade all server-side components together, you do not have to upgrade client components at the same time. A Jolt 1.1 client and Jolt 1.2 server interoperate, but the client has only 1.1 functionality; server components have additional functionality. However, a Jolt 1.2 client cannot operate with a Jolt 1.1 server.

- **ASP Connectivity for BEA Tuxedo** — The name of Jolt Web Application Server (WAS) has changed to ASP Connectivity for BEA Tuxedo to emphasize the Microsoft ASP environment. Also, ASP Connectivity for BEA Tuxedo is no longer an add-on; it is part of the Jolt Client Class Library.

- **JSE Connectivity for BEA Tuxedo** — The name of Jolt WAS for Servlets is now JSE Connectivity for BEA Tuxedo.

- **JoltBeans** — This feature is no longer an add-on; it is part of the Jolt Client Class Library.

- **Security/Encryption** — Diffie-Hellman (DH) key exchange replaces the DES key exchange mechanism (for U. S. customers only).

- **Digital Alpha NT** — Is no longer supported.

System Requirements

The following hardware and software components are required before you can install Jolt.
WebLogic Enterprise

If you plan to use Jolt with WebLogic Enterprise, you must have WebLogic Enterprise 5.1 installed before you install Jolt.

Web Servers Supported

To provide HTTP/HTML-based access to enterprise applications, Jolt supports the Microsoft Internet Information Server (IIS) or Java Web Server.

Supported Platforms for Jolt Server

Jolt server platform support depends upon BEA Tuxedo version support, as stated earlier in this chapter. The Jolt server also requires:

- CD-ROM access
- 500K of disk space

Table 14-1 shows the supported platforms for the Jolt server.

<table>
<thead>
<tr>
<th>Operating System Vendor</th>
<th>Operating System</th>
<th>Operating System Version</th>
<th>BEA Tuxedo Version Supported</th>
<th>WebLogic Enterprise Version Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Digital Equipment Corporation (DEC)</td>
<td>Digital UNIX</td>
<td>4.0d</td>
<td>6.5 only</td>
<td></td>
</tr>
<tr>
<td>DEC</td>
<td>Digital UNIX</td>
<td>4.0e</td>
<td>4.2</td>
<td></td>
</tr>
<tr>
<td>Hewlett-Packard (HP)</td>
<td>HP-UX</td>
<td>10.20</td>
<td>6.4 and 6.5</td>
<td>4.2</td>
</tr>
<tr>
<td>HP</td>
<td>HP-UX</td>
<td>11.0</td>
<td>6.4 and 6.5</td>
<td>4.2</td>
</tr>
</tbody>
</table>
### Table 14-1 Supported Platforms for Jolt Server (Continued)

<table>
<thead>
<tr>
<th>Operating System Vendor</th>
<th>Operating System</th>
<th>Operating System Version</th>
<th>BEA Tuxedo Version Supported</th>
<th>WebLogic Enterprise Version Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>International Business Machines (IBM)</td>
<td>AIX</td>
<td>4.2.1 (RS/6000)</td>
<td>6.4 and 6.5</td>
<td></td>
</tr>
<tr>
<td>IBM</td>
<td>AIX</td>
<td>4.3.1 (RS/6000)</td>
<td>6.4 and 6.5</td>
<td></td>
</tr>
<tr>
<td>IBM</td>
<td>AIX</td>
<td>4.3.2</td>
<td>4.2</td>
<td></td>
</tr>
<tr>
<td>IBM</td>
<td>OS/390</td>
<td>V2R6</td>
<td>6.5 only</td>
<td></td>
</tr>
<tr>
<td>IBM</td>
<td>OS/400</td>
<td>4.1 (RISC)</td>
<td>6.5 only</td>
<td></td>
</tr>
<tr>
<td>Microsoft</td>
<td>Windows NT</td>
<td>4.0 (sp4)</td>
<td>6.4 and 6.5</td>
<td>4.2</td>
</tr>
<tr>
<td>Red Hat</td>
<td>Linux</td>
<td>5.2</td>
<td>6.5 only</td>
<td></td>
</tr>
<tr>
<td>Santa Cruz Operation (SCO)</td>
<td>Unixware</td>
<td>7.0</td>
<td>6.5 only</td>
<td></td>
</tr>
<tr>
<td>Sequent</td>
<td>Dynix/PTS</td>
<td>4.4.2</td>
<td>6.5 only</td>
<td>4.2</td>
</tr>
<tr>
<td>SGI (Silicon Graphics)</td>
<td>IRIX</td>
<td>6.4 -o32</td>
<td>6.4 and 6.5</td>
<td></td>
</tr>
<tr>
<td>SGI</td>
<td>IRIX</td>
<td>6.5 -n32</td>
<td>6.5 only</td>
<td>4.2</td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.5.1 (SPARC)</td>
<td>6.4 and 6.5</td>
<td></td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.5.1 (Intel)</td>
<td>6.4 and 6.5</td>
<td></td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.6 (SPARC)</td>
<td>6.4 and 6.5</td>
<td>4.2</td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.6 (Intel)</td>
<td>6.4 and 6.5</td>
<td></td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.7 (SPARC)</td>
<td>6.5 only</td>
<td>4.2</td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.7 (Intel)</td>
<td>6.5 only</td>
<td></td>
</tr>
<tr>
<td>Siemens Nixdorf (SNI)</td>
<td>Reliant UNIX</td>
<td>5.4.4b</td>
<td>6.5 only</td>
<td></td>
</tr>
</tbody>
</table>
Supported Platforms for Jolt Relay

Table 14-2 shows the supported platforms for Jolt Relay:

<table>
<thead>
<tr>
<th>Operating System Vendor</th>
<th>Operating System</th>
<th>Operating System Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEC</td>
<td>Digital Tru64 UNIX</td>
<td>4.0f (Alpha)</td>
</tr>
<tr>
<td>HP</td>
<td>HP-UX</td>
<td>11.00</td>
</tr>
<tr>
<td>IBM</td>
<td>AIX</td>
<td>4.3.3</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Windows NT</td>
<td>4.0 (Intel) (sp4)</td>
</tr>
<tr>
<td>Sun</td>
<td>Solaris</td>
<td>2.6 (SPARC)</td>
</tr>
</tbody>
</table>

Client Support

Jolt 1.2 supports Java applets and standalone Java client applications for BEA Tuxedo. The following table shows the client types supported by Jolt 1.2.

<table>
<thead>
<tr>
<th>Client Type</th>
<th>Supports</th>
<th>Version Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Java Applet in a Browser</td>
<td>Microsoft Internet Explorer</td>
<td>Internet Explorer (IE) 4.0, 5.0</td>
</tr>
<tr>
<td></td>
<td>Netscape</td>
<td>Netscape Communicator 4.7</td>
</tr>
<tr>
<td>Standalone Applications on a Desktop</td>
<td>Java</td>
<td>JDK 1.1.7, 1.2.2</td>
</tr>
</tbody>
</table>

The HTML-based Jolt Client Classes run inside a Web server. The following Web servers are certified with Jolt.
Jolt Client Requirements

Jolt has the following client requirements:

- 574K of disk space for client classes.
- 1364K of disk space for client API documentation.
- 190K of disk space for client examples.
- Java Developer’s Kit (JDK) 1.1.7 or 1.2.2 is certified for Jolt 1.2 application development. Jolt 1.1 uses JDK 1.1.5 or 1.1.6. (http://java.sun.com/java.sun.com/products/JDK/index.html)
- Java-enabled browser (Internet Explorer) or Java Virtual Machine (JVM).
Jolt Client Class Library

Various implementations of Java tend to show minor differences in characteristics. Jolt 1.2 is based on the 1.1.7 JDK. The Jolt 1.2 class library is compatible with JDK versions 1.1.7 and 1.2.2 on the following operating systems:

- Solaris 2.5.1, 2.6, and 2.7
- Windows NT 4.0
- Windows 95 and 98

The Jolt class library is compatible with the browsers and JDK versions shown in Table 14-5.

**Table 14-5  Jolt Class Library Compatibility**

<table>
<thead>
<tr>
<th>Vendor</th>
<th>Browser/Version</th>
<th>Java Virtual Machine (JVM)</th>
<th>OS Version</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft</td>
<td>Internet Explorer 4.0.1 (sp1)</td>
<td>MS JVM 4.0</td>
<td>Windows NT 4.0 Service Pack 4</td>
</tr>
<tr>
<td>Microsoft</td>
<td>Internet Explorer 5.0</td>
<td>MS JVM 4.0</td>
<td>Windows NT 4.0 Service Pack 4</td>
</tr>
<tr>
<td>Netscape</td>
<td>Communicator 4.7</td>
<td>JDK 1.1.5</td>
<td>Windows NT 4.0</td>
</tr>
</tbody>
</table>

Release Migration/Interoperability

If you have Jolt 1.1 installed, we recommend that you uninstall it before you install Jolt 1.2.

Jolt provides increased interoperability between versions. In the Jolt 1.2 release, however:

- You must upgrade all server-side components together.
If you have Jolt 1.1 applications, only Jolt 1.1 functionality is available to the Jolt 1.1 client, even though the server-side components have additional functionality when they are upgraded to Jolt 1.2.

A Jolt 1.2 client cannot operate with a Jolt 1.1 server.

Jolt ASP Connectivity for BEA Tuxedo

Requirements

Before you install Jolt ASP Connectivity for BEA Tuxedo, you must have the following software installed:

- Windows NT Server 4.0 with NT Option Pack
- Microsoft IIS 4.0
- WebLogic Enterprise 5.1

Installation Instructions

Note: These instructions are documented from the Microsoft standard. Check the Microsoft documentation for updates.

To install Jolt ASP Connectivity for BEA Tuxedo:

1. Remove any files that you copied when you installed the Jolt 1.1 WAS package.
2. Install the Microsoft Java Component Framework from the IIS 4.0 SDK (if it is not already installed).
3. Create a new directory, aspcomp in the java\Trustlib directory. (This is most typically found in %windir%\java\TrustLib).

Since the framework files have been created in a package called aspcomp, the Java Virtual Machine (JVM) expects to find them there.
Preparing to Install BEA Jolt

The Microsoft Java Component Framework files are typically found in:
C:\InetPub\iissamples\sdk\components\java\Framework

4. Copy the framework class files to the newly created
   java\TrustLib\aspcomp directory.
   (You only need to copy the *.class files.)

5. Install the latest Jolt 1.2 classes in the Windows NT server java\Trustlib directory.
   Contact BEA Customer Support for the latest Jolt 1.2 patch files.

6. Copy the class subdirectories and files (or unzip jolt.zip) from the Jolt 1.2 for
   WebLogic Enterprise 5.1 client distribution into your java\lib directory. The
   Jolt 1.2 classes have been created in packages beginning with “bea” so the JVM
   expects to find them in the java\lib\bea\... subdirectories.
   Install the corresponding Jolt 1.2 patch files on your BEA Tuxedo server if you
   have not already done so.

7. Install the Web Application Services classes in the Windows NT server
   java\Trustlib directory.

8. Unzip joltjsp.jar to create the class files in your java\trustlib directory.
   Make sure you expand the files using the subdirectory folders in the zip archive.
   The Web Application Services classes are created in packages beginning with
   “bea” so the JVM will expect to find them in the java\Trustlib\bea\... subdirectories.

9. Register the Web Application Service classes as ActiveX components.
   Run the wasreg.cmd command file to register the BEA Web Application
   Services java classes as BEAWEB ActiveX components. This enables the
   BEAWEB components to be accessible from the Microsoft ASP scripts.

wasreg.cmd file list
@echo off
REM
REM This batch command file registers the necessary java
REM classes as ActiveX components so that they are accessible
REM from Active Server Pages or any other ActiveX client.
REM The classes (or equivalent zip file) must exists
REM in your java/Trustlib directory (normally C:\%WINDIR%\java\Trustlib.

14-10 Installation Guide
REM
javareg /q /register /class:bea.jolt.pool.asp.UserInfo/progid:BEAWEB.UserInfo
javareg /q/register/class:bea.jolt.pool.asp.SessionPoolManager/progid:BEAWEB.SessionPoolManager
javareg /q /register /class:bea.jolt.pool.asp.Template/progid:BEAWEB.Template
javareg /q /register /class:bea.jolt.pool.asp.TemplateData/progid:BEAWEB.TemplateData

REM
REM These are the new ActiveX component names.
REM
javareg /q /register /class:bea.jolt.pool.asp.UserInfo/progid:BEAJOLTPOOL.AspUserInfo
javareg /q /register /class:bea.jolt.pool.asp.AspPoolManagerConfig/progid:BEAJOLTPOOL.AspPoolManagerConfig
javareg /q /register /class:bea.jolt.pool.asp.AspSessionPoolManager/progid:BEAJOLTPOOL.AspSessionPoolManager
javareg /q /register /class:bea.jolt.pool.asp.AspTemplate/progid:BEAJOLTPOOL.AspTemplate
javareg /q /register /class:bea.jolt.pool.asp.AspDataSet/progid:BEAJOLTPOOL.AspDataSet

REM
REM These are the new ActiveX component names.
REM
javareg /q /register /class:bea.jolt.pool.asp.UserInfo/progid:BEAJOLTPOOL.AspUserInfo
javareg /q /register /class:bea.jolt.pool.asp.AspPoolManagerConfig/progid:BEAJOLTPOOL.AspPoolManagerConfig
javareg /q /register /class:bea.jolt.pool.asp.AspSessionPoolManager/progid:BEAJOLTPOOL.AspSessionPoolManager
javareg /q /register /class:bea.jolt.pool.asp.AspTemplate/progid:BEAJOLTPOOL.AspTemplate
javareg /q /register /class:bea.jolt.pool.asp.AspDataSet/progid:BEAJOLTPOOL.AspDataSet

References

Refer to the online Microsoft NT Option Pack Product Documentation, especially the Microsoft Internet Information Server (IIS) chapters.
Preinstallation Checklist

Before you install Jolt, check the following:

- Back up your existing CATNAMES, jrepository and jrly.config files.
- Uninstall previous version of Jolt, if applicable. Note that you can install all Jolt 1.2 components at once or separately, in the same or different directories. However, a single uninstall removes everything: files, directories and registry entries.
- Verify that you have WebLogic Enterprise 5.1 installed, if applicable.
- Determine the location of the BEA Tuxedo directory where the Jolt server is to be installed.
- Determine the location of the directory where the Jolt documentation is to be installed.
- Determine the Web server location where the Jolt client components are to be installed.
- Verify the user ID and group ID assigned to Jolt server files (on UNIX).
- Verify the user ID and group ID to be assigned to Jolt client files (on UNIX).

Caution: Jolt 1.2 automatically installs two Microsoft dynamic link libraries (DLLs), MSVCRT.DLL and MFC42U.DLL and overwrites older versions of these libraries. Before you begin installation, check whether older versions of these dynamic link libraries already exist. If they do exist and you do not want them to be overwritten, back them up.
CHAPTER 15 Installing BEA Jolt

This chapter explains how to install the BEA Jolt software.

This chapter includes the following sections:

- Microsoft Windows NT Installation Instructions
- UNIX System Installation Instructions
- Licensing Jolt for WebLogic Enterprise 5.1

Microsoft Windows NT Installation Instructions

The installation (setup.exe) is launched automatically when you insert the Jolt Installation CD and provides a set of step-by-step installation windows to help you quickly install your Jolt product. These windows are self-explanatory.

You can cancel the installation at any time. (If your system detects the presence of a previous Jolt installation, you are given the option of aborting the current installation or overwriting the existing one.)

1. When you see the Welcome window shown in Figure 15-1, click the Next button to proceed with the installation.
2. The next window that appears is the **Software License Agreement** (Figure 15-2). Use the scroll bar or the **Page Down** key to read the Software License Agreement.

To continue with the Jolt installation, you must accept the terms of the license agreement. If you accept the terms, click the **Yes** button to continue with the installation. If you do not accept the terms, click the **No** button and the installation stops.
3. To install Jolt, you must have either WebLogic Enterprise 5.1 already installed. If you have both BEA Tuxedo and WebLogic Enterprise installed, the Select Package window (Figure 15-3) appears.

Click either the BEA Tuxedo or WebLogic Enterprise radio button to select the base on which to install Jolt.
4. If you have either BEA Tuxedo or WebLogic Enterprise installed, the Select Components window appears (Figure 15-4).

In the Select Components window, select the module(s) you want to install. (You can choose to install any or all components.) The sample window shown in Figure 2-4 displays BEA Tuxedo components. If you install on WebLogic Enterprise, the server component listed would show “Jolt Server for WLE” instead of “Jolt Server for Tuxedo.”

Note: Selection in this window works as a toggle. To make your selection, click in the check box to the left of the text representing your choice. The window displays a checkmark to the left of your selection. To deselect a component, click on the checkmark to the left of your choice and the checkmark is removed.

Table 15-1 displays what Jolt installs with each selected component.

You can check the available and required disk space on a particular drive either in the Space Available section at the bottom of the Select Component window,
or by clicking on **Disk Space** in the same window. The **Space Required** number is approximately equal to the space required by the total number of components you select plus a core component (uninstall files).

The **Destination Folder** displays the directory where the Jolt components are installed. To change the directory, click the **Browse** button and type the directory path. You have another opportunity to specify the Destination Folder when you select the Jolt components for installation (Step 5).

**Note:** Typically, the destination path is the BEA Tuxedo or WebLogic Enterprise directory.

Click the **Next** button to continue with the installation.

**Figure 15-4  Select Components Window**

Table 15-1 describes what gets installed with each component.
Installing BEA Jolt

Table 15-1 What Gets Installed with Each Component

<table>
<thead>
<tr>
<th>Component...</th>
<th>Installs...</th>
</tr>
</thead>
<tbody>
<tr>
<td>Client API Documentation</td>
<td>Javadoc Language Reference documentation only</td>
</tr>
<tr>
<td>Client Examples</td>
<td>Client Examples only</td>
</tr>
<tr>
<td>Client Classes</td>
<td>Client classes (including JoltBeans) only</td>
</tr>
<tr>
<td>Jolt Server for WebLogic Enterprise</td>
<td>Jolt server only (review license)</td>
</tr>
<tr>
<td>Jolt Relay Front-End</td>
<td>Jolt Internet Relay front-end server</td>
</tr>
<tr>
<td>Jolt Relay Back-End</td>
<td>Jolt Internet Relay back-end server</td>
</tr>
</tbody>
</table>

5. A Component Destination window appears for each component you have chosen to install, that is, each component that you checked in the Select Component window. (Figure 15-5 displays the Server Destination window.) In each Component Destination window, check that the Destination Folder is where you want the component installed. If it is not, click on the Browse button and change it.

Click on the Next button.
6. A prompt (Figure 15-6) asks if you have Jolt 1.1 applications.

   If you click the Yes button, in addition to the client class .jar files, you also get client classes in .zip file format and all Jolt client classes in the <classes/bea/jolt> directory.

   If you click the No button, you get only Jolt client class .jar files.

**Figure 15-6  Jolt 1.1 Applications**
7. After the selected modules are installed, you are prompted (Figure 15-7) to install your Jolt license.

If you click the Yes button, the **Insert License Disk** window (Figure 2-8) displays.

If you click the **No** button, a warning reminds you that your license has not been installed. For further information on licensing, see the “Licensing Your Jolt Software” section later in this chapter.

Figure 15-7  **Install License Window**

<table>
<thead>
<tr>
<th>Question</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Question" /></td>
</tr>
<tr>
<td><strong>Would you like to install your license now?</strong></td>
</tr>
<tr>
<td><img src="image" alt="Yes" /></td>
</tr>
</tbody>
</table>

8. If you choose to install your license, the **Insert License Disk** window shown in Figure 15-8 displays and you are prompted to insert your license disk.

Type the designated path or click the **Browse** button to change the drive, then click the **OK** button.

Figure 15-8  **Insert License Disk Window**

<table>
<thead>
<tr>
<th>Insert License Disk</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Insert License Disk" /></td>
</tr>
<tr>
<td><strong>Please insert your BEA Jolt License Disk</strong></td>
</tr>
<tr>
<td><strong>Path:</strong></td>
</tr>
<tr>
<td><img src="image" alt="Path" /></td>
</tr>
<tr>
<td><img src="image" alt="OK" /></td>
</tr>
</tbody>
</table>
9. If you chose to install the Jolt Relay Front End in the Select Component window (Figure 15-4), a prompt asks if you want to install Jolt Relay as a service (Figure 15-9). If you click on the Yes button, the Enter Information (Figure 15-10) window appears. If you click on the No button, the Setup Complete (Figure 15-11) window appears.

Figure 15-9  Install Jolt Relay Window
10. Enter the path for your configuration file. Click on the **Next** button.

**Figure 15-10**  The Enter Information Window

11. When all files are installed, the **Setup Complete** (Figure 15-11) window displays.

   Click on **Finish**. Jolt is installed.
The Jolt installation shell script for UNIX and Linux systems includes all components necessary for installing the Jolt system: the Jolt Repository, the Jolt Server, the Jolt Relay front-end and back-end, and the Jolt Class Library.
Invoking the UNIX Installation Script

Before you install Jolt, make sure that WebLogic Enterprise 5.1 has been installed. (If your system detects the presence of a previous Jolt installation, you are given the option of aborting the current installation or overwriting the existing one.)

1. Log in as a user who has write permission in the BEA Tuxedo directory.

2. Insert the CD-ROM in the CD-ROM drive. If you are running on Solaris and the daemon /usr/sbin/vold is running, the CD-ROM should be automatically mounted in the /cdrom/JOLT directory.

   cd /cdrom/jolt/unix

   If you are not running on Solaris or vold is not running, consult your UNIX administration documentation to mount the CD-ROM.

3. Type ls

   The directory contents should look similar to the following sample. If not, verify that you are installing the correct CD-ROM.

   hp/ibm/
   install.sh
   /sun5x/alpha

4. Type install.sh

5. Press Enter.

   This invokes the Jolt installation script. The step-by-step install screens are described in the following section.

Unix/Linux System Installation Script

The UNIX system installation script provides a set of step-by-step instructions to help you quickly install your Jolt product. This script lets you specify your platform, operating system, and other installation details. The installation script prompts you through the entire installation process. You can cancel the installation at any time by pressing CTRL-C simultaneously.

Note: The script used to show the UNIX installation is taken from Jolt 1.2 for WebLogic Enterprise 5.1.
The following installation options are available:

1 WLE Install Jolt for WLE

Select an option (default: Tuxedo) [?,?,?,q]: 1

01) alpha/tru64/Wle5.1 02) hp/hpux11/Wle5.1
03) ibm/aix43/Wle5.1 04) sun/sol26/Wle5.1

Install which platform’s files? [01-04, q to quit, 1 for list]: 04

BEA Jolt Release 1.2

This directory contains the BEA Jolt System for
SunOS 5.6 (Solaris 2.6) on SPARC

Is this correct? [y,n,q]: y

To terminate the installation at any time press the interrupt key, typically <del>, <break>, or <ctrl+c>.

The following packages are available:

1 jolt BEA Jolt

Select the package(s) you wish to install (or ‘all’ to install all packages) (default: all) [?,?,?,q]: 1

BEA Jolt (sparc) Release 1.2
Copyright (c) 1997-1999 BEA Systems, Inc.
Portions * Copyright 1986-1997 RSA Data Security, Inc.
All Rights Reserved.
TUXEDO, Jolt, and WebLogic are registered trademarks of BEA Systems, Inc.
BEA WebLogic Enterprise is a trademark of BEA Systems, Inc.

The following installation options are available:

1 all Install Jolt server, relay, rad, and client
2 server Install the server only
Installing BEA Jolt

3 relay  Install the relay front-end only
4 rad    Install the relay back-end only
5 client Install the client only
6 compat Install the Jolt 1.1 Client compatibility classes

Select an option (default: all) [?,??,q]: 1

Note that the jolt server will be installed into an existing TUXEDO or WLE directory. You MUST have previously installed TUXEDO version 6.4 or 6.5 or WLE 4.2 to attempt this installation.

Base directory of existing TUXEDO installation [?,q]: /usr/jolt/W5.1u

Determining if sufficient space is available ...
3068 blocks are required
1045034 blocks are available to /usr/jolt/W5.1u

Using /usr/jolt/W5.1u as the TUXEDO base directory

Unloading
/cdrom/cdrom0/jolt/unix/sun5x/sol26/Wle5.1/jolt/joltc1t.2 ...
udataobj/jolt/jolt.jar
udataobj/jolt/joltasp.jar
udataobj/jolt/joltjse.jar
udataobj/jolt/JoltBeanDev.jar
udataobj/jolt/JoltBeanDevAwt.jar
udataobj/jolt/JoltBeanDevSwing.jar
udataobj/jolt/JoltBeanDevSwing11.jar
udataobj/jolt/JoltBeanRt.jar
udataobj/jolt/JoltBeanRtAwt.jar
udataobj/jolt/JoltBeanRtSwing.jar
udataobj/jolt/JoltBeanRtSwing11.jar
udataobj/jolt/RE.html
udataobj/jolt/examples/asp/bankapp/templates/deposit.temp
udataobj/jolt/examples/asp/bankapp/templates/inquiry.temp
udataobj/jolt/examples/asp/bankapp/templates/transfer.temp
udataobj/jolt/examples/asp/bankapp/templates/withdrawal.temp
udataobj/jolt/examples/asp/bankapp/templates/error.temp
udataobj/jolt/examples/asp/bankapp/templates/nosession.temp
udataobj/jolt/examples/asp/bankapp/templates/syseerror.temp
udataobj/jolt/examples/asp/bankapp/bankapp.properties
UNIX System Installation Instructions

udataobj/jolt/examples/asp/bankapp/depositForm.htm
udataobj/jolt/examples/asp/bankapp/inquiryForm.htm
udataobj/jolt/examples/asp/bankapp/transferForm.htm
udataobj/jolt/examples/asp/bankapp/withdrawalForm.htm
udataobj/jolt/examples/asp/bankapp/tellerForm.asp
udataobj/jolt/examples/asp/bankapp/tlr.asp
udataobj/jolt/examples/asp/bankapp/web_templates.inc
udataobj/jolt/examples/asp/bankapp/web_admin.inc

udataobj/jolt/classes/bea/jolt/ServiceException.class
udataobj/jolt/classes/bea/jolt/Session.class
udataobj/jolt/classes/bea/jolt/SessionException.class
udataobj/jolt/classes/bea/jolt/SessionState.class
udataobj/jolt/classes/bea/jolt/Transaction.class
udataobj/jolt/classes/bea/jolt/TransactionException.class
udataobj/jolt/classes/bea/jolt/UrlHdlr.class
udataobj/jolt/classes/bea/jolt/UrlInfo.class
udataobj/jolt/jolt.zip

910 blocks
... finished

Unloading /cdrom/cdrom0/unix/sun5x/sol26/Tux6.4_6.5/jolt/joltrad.Z

... bin/JRAD
locale/CATNAMES
locale/C/JRAD_CAT
locale/C/JRAD.text
140 blocks

... finished

Unloading /cdrom/cdrom0/unix/sun5x/sol26/Tux6.4_6.5/jolt/joltsvr.Z

... bin/JREPSVR
bin/JSL
bin/JSH
lib/libjconv.so
include/jotypes.h
locale/CATNAMES
locale/C/JOLT_CAT
locale/C/JOLT.text
udataobj/jrep.f32
udataobj/jwsldapm.f32
udataobj/jolt/repository/jrepository
760 blocks
... finished
Please don’t forget to manually append the contents of the license file from the enclosed floppy to ‘/usr/jolt/W5.1u/udataobj/lic.txt’. Refer to the BEA Jolt manual for details on how to do this.

Changing file permissions...
... finished

Installation of BEA Jolt was successful

The following packages are available:

1 jolt BEA Jolt

Select the package(s) you wish to install (or ‘all’ to install all packages) (default: all) [?,??,q]: q

### Licensing Jolt for WebLogic Enterprise 5.1

The licensing scheme used by Jolt depends on the version of BEA Tuxedo used with Jolt. You need the following information to install your license:

- The version of BEA Tuxedo you are installing.
- Your BEA Tuxedo directory (`TUXDIR`) from the installation. This is the directory that contains the BEA Tuxedo directories (`bin`, `udataobj`, etc.).
- Your license file

Jolt for WebLogic Enterprise 5.1 uses a digitally signed license file to enable a license. This file is provided on a floppy disk shipped with your Jolt software.

The UNIX installation program (`install.sh`) does not install the license automatically; see the “UNIX Licensing Instructions” in the following section.

The NT installation program (`setup.exe`) prompts you for the location of the Jolt license file. When you provide the necessary information, the installation program installs the license file for you. If you do not install the license file during installation, follow the steps in the “NT Licensing Instructions” section.
UNIX Licensing Instructions

1. Identify your current BEA Tuxedo license file. This is located in $TUXDIR/udataobj/lic.txt.

2. Make a copy of this file:
   cd $TUXDIR/udataobj
   cp lic.txt lic.txt.bak

3. Check that you have completed Step 2. Verify the copy using OS-specific commands (e.g., `diff` on UNIX systems).

4. Append the contents of the Jolt license file to the BEA Tuxedo license file:
   cat /dev/diskette/joltlic.txt >> lic.txt

NT Licensing Instructions

1. Identify your current BEA Tuxedo license file. This is located in %TUXDIR%\udataobj\lic.txt.

2. Make a copy of this file:
   cd %TUXDIR%\udataobj
   copy lic.txt lic.txt.bak

3. Check that you have completed Step 2. Verify the copy using OS-specific commands.

4. Append the contents of the Jolt license file to the BEA Tuxedo license file:
   copy lic.txt + a:\joltlic.txt

   A text editor can be used to copy and paste the contents of the Jolt license file into the BEA Tuxedo license file.

   **Note:** The digital signature is 64 characters long. Every character must match exactly or the license is not valid.
This chapter explains how to configure BEA Jolt. It contains a “Quick Configuration” section for users who are familiar with Jolt. The rest of the chapter provides more detailed information. It is presumed that readers of this chapter are system administrators or application developers who have experience with the operating systems and workstation platforms on which they are configuring Jolt.

This chapter includes the following sections:

- Quick Configuration
- Jolt Background Information
- Jolt Relay
- Jolt Relay Adapter
- Jolt Repository
- Event Subscription
- BEA Tuxedo Background Information
- Sample Applications in Jolt Online Resources
Quick Configuration

If you are already familiar with Jolt and BEA Tuxedo, this section provides a quick guide to the configuration procedure. If you have not used Jolt before, read the “Jolt Background Information” section in this chapter.

Configure Jolt on BEA Tuxedo

Follow the directions in this section to configure the Jolt Server Listener (JSL).

Edit the UBBCONFIG file

1. In the MACHINES Section, specify MAXWSCLIENTS=number (Required).

   **Note:** If MAXWSCLIENTS is not set, JSL does not boot.

2. In the GROUPS section, set GROUPNAME <required parameters> [optional parameters].

3. Set the SERVERS section (Required).

   Lines within this section have the form:

   JSL <required parameters> [optional parameters]

   where JSL specifies the file (string_value) to be executed by tmboot(1).

4. Set the required parameters for JSL.

   Required parameters are:

   SVRGRP, SVRID, and CLOPT="-A...-n...<host port>"

5. Set other parameters for JSL.

   The following parameters can be used with the JSL, but you need to understand how doing so would affect your application. See “Other Parameters Usable With JSL” later in this chapter.

   SVRGRP=string_value
   SRVID=number
Configure the Jolt Repository

In the GROUPS section:

1. Specify the same identifiers given as the value of the LMID parameter in the MACHINES section.

2. Specify the value of the GRPNO, between 1 and 30,000.

In the SERVERS section:

The Jolt Repository Server (JREPSVR) contains services for accessing and editing the Repository. Multiple JREPSVR instances share repository information through a shared file. Include JREPSVR in the SERVERS section of the UBBCONFIG file.

1. Indicate a new server identification with the SRVID parameter.

2. Specify the -W flag for one (and only one) JREPSVR to ensure that you can edit the Repository. (Without this flag, the Repository is read-only.)

3. Type the -P flag to specify the path of the repository file. (An error message displays in the BEA Tuxedo ULOG file if the argument for the -P flag is not entered.)

4. Add the file pathname of the Repository file (for example, /app/jrepository).

5. Boot the BEA Tuxedo system using the tmloadcf and tmboot commands.

Initialize Services Using BEA Tuxedo and the Repository Editor

You must initially define the BEA Tuxedo services using BEA Tuxedo and Jolt in order to make the Jolt services available to the client.

1. Build the BEA Tuxedo server that contains the service.

2. Access the Jolt Repository Editor.
Before You start the Repository Editor

Before you start the Repository Editor, make sure that you have installed all necessary Jolt software. You cannot use the Repository Editor until JREPSVR and JSL are running.

Start the Repository Editor from either the JavaSoft appletviewer or from your Web browser.

Starting the Repository Editor Using appletviewer:

1. Set the CLASSPATH to include the Jolt class directory or the directory where the *.jar files reside.
2. If loading the applet from a local disk, type the following at the URL location:
   
   `appletviewer <full-pathname>/RE.html`

   If loading the applet from the Web server, type the following at the URL location:

   `http://<www.server>/<URL path>/RE.html`

3. Press Enter. The Repository Editor logon window displays.

Starting the Repository Editor from a Web server:

To start the Repository Editor from a local file:

1. Set the CLASSPATH to include the Jolt class directory.
2. Type: `file:<full-pathname>/RE.html`

To start the Repository Editor from a Web server:

1. Ensure that the CLASSPATH does not include the Jolt class directory
2. Remove the Jolt classes from CLASSPATH.
3. Type the following:

   `http://<www.server>/<URL path>/RE.html`
**Note:** If jolt.jar and admin.jar are in the same directory as RE.html, the web server provides the classes. If they are not in the same directory as RE.html, modify the applet code base.

4. Press **Enter**. The Repository Editor logon window displays.

**Log on to the Repository Editor**

After starting the Jolt Repository Editor, follow these directions to log on:

1. Type the name of the server machine designated as the “access point” to the BEA Tuxedo application and select the port number text field.

2. Type the port number and press **Enter**. The system validates the server and port information.

   **Note:** Unless you are logging on through the Jolt Relay, the same port number is used to configure the Jolt Listener. Refer to your UBBCONFIG file for additional information.

3. Type the BEA Tuxedo Application Password and press **Enter**. Based on the authentication level, type the remaining information.

4. Type the BEA Tuxedo user name and press **Tab**.

5. Type the BEA Tuxedo user password and press **Enter**.

   **Note:** The Jolt 1.2 Repository Editor uses the hardcoded joltadmin for the user role.

   The **Packages** and **Services** options are activated.
Repository Editor Logon Window Description

The following table details the options of the Repository Editor logon window.

<table>
<thead>
<tr>
<th>Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Server</td>
<td>Type the server name.</td>
</tr>
<tr>
<td>Port Number</td>
<td>Type the port number in decimal value.</td>
</tr>
<tr>
<td>Note</td>
<td>After the server name and port number are entered, the user name and password fields are activated. Activation is based on the authentication level of the BEA Tuxedo application.</td>
</tr>
<tr>
<td>Application Password</td>
<td>BEA Tuxedo administrative password text entry.</td>
</tr>
<tr>
<td>User Name</td>
<td>BEA Tuxedo user identification text entry. The first character must be an alpha character.</td>
</tr>
<tr>
<td>User Password</td>
<td>BEA Tuxedo password text entry.</td>
</tr>
<tr>
<td>User Role</td>
<td>BEA Tuxedo user role. Required only if BEA Tuxedo authentication level is USER_AUTH or higher.</td>
</tr>
<tr>
<td>Packages</td>
<td>Accesses the Packages window. (Enabled after the logon.)</td>
</tr>
<tr>
<td>Services</td>
<td>Accesses the Services window. (Enabled after the logon.)</td>
</tr>
</tbody>
</table>

**Exit the Repository Editor**

Exit the Repository Editor when you are finished adding, editing, testing, or deleting packages, services, and parameters. Figure 16-1 is an example of the Repository Editor window before exiting. Only Packages, Services, and Close are enabled. All text entry fields are disabled.
Figure 16-1  Example of the Repository Editor Logon Window Before Exiting

![Repository Editor Logon Window](image)

To exit the Repository Editor:

1. Select **Back** from a previous window to return to the Logon window.
2. Select **Close** to terminate the connection with the server. The Repository Editor Logon window continues to display with disabled fields.
3. Select **Close** from your browser menu to remove the window from your screen.

**Configure the BEA Tuxedo TMUSREVT Server for Event Subscription**

Jolt Event Subscription is used to receive event notifications from either BEA Tuxedo services or other BEA Tuxedo clients. Configure the BEA Tuxedo TMUSREVT server and modify the application UBBCONFIG file. Listing 16-1 shows the relevant TMUSREVT parameters in the UBBCONFIG file:
Configuring the BEA Jolt System

Listing 16-1  TMUSREVT Parameters in the UBBCONFIG File

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>TMUSREVT</td>
<td>SRVGRP=EVBGRP1  SRVID=40</td>
</tr>
<tr>
<td></td>
<td>GRACE=3600</td>
</tr>
<tr>
<td>ENVFILE</td>
<td>&quot;/usr/tuxedo/bankapp/TMUSREVT.ENV&quot;</td>
</tr>
<tr>
<td>CLOPT</td>
<td>&quot;-e tmusrevt.out -o tmusrevt.out -A -- -f /usr/tuxedo/bankapp/tmusrevt.dat&quot;</td>
</tr>
<tr>
<td>SEQUENCE</td>
<td>11</td>
</tr>
</tbody>
</table>

In the SERVERS sections of the UBBCONFIG file, specify the SRVGRP and SRVID.

Configure Jolt Relay

On UNIX:

Start the JRLY process on UNIX by typing the following command at the system prompt:

```
jrly -f <config_file_path>
```

If the configuration file does not exist or cannot be opened, the JRLY writes a message to standard error, attempts to log the startup failure in the error log, then exits.

On UNIX and NT:

The format of the configuration file is a TAG=VALUE format. Blank lines or lines starting with a "#" are ignored. Listing 16-2 is an example of the formal specifications of the configuration file.

Listing 16-2  Formal Configuration File Specifications

```
LOGDIR=<LOG_DIRECTORY_PATH>
ACCESS_LOG=<ACCESS_FILE_NAME in LOGDIR>
ERROR_LOG=<ERROR_FILE_NAME in LOGDIR>
LISTEN=<IP:Port combination where JRLY will accept comma-separated connections>
CONNECT=<IP:Port1, IP:Port2...IP:PortN:Port (List of IP:Port combinations associated with JRADs: can be 1...N)>
```
Quick Configuration

On NT Only (Optional):

`SOCKETTIMEOUT` is the time in seconds for which JRLY NT service blocks for network activity (new connections, data to be read, closed connections). `SOCKETTIMEOUT` also affects the SCM. When the SCM requests the NT service to stop, the SCM must wait for at least `SOCKETTIMEOUT` seconds before quitting.

**Note:** The format for directory and file names is determined by the operating system. UNIX systems use the forward slash (`/`). NT systems use the backslash (`\`). If any of the files specified in `LOGDIR`, `ACCESS_LOG`, or `ERROR_LOG` cannot be opened for writing, the JRLY prints an error message on `stderr` and exits.

The format for host names and port numbers are shown in Table 16-1.

### Table 16-1 Host Name and Port Number Formats

<table>
<thead>
<tr>
<th>Host Name/Port Number</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hostname:Port</td>
<td>Hostname is a string; Port is a decimal number.</td>
</tr>
<tr>
<td>//Hostname:Port</td>
<td>Hostname is a string; Port is a decimal number.</td>
</tr>
<tr>
<td>IP:Port</td>
<td>IP is a dotted notation IP address; Port is a decimal number.</td>
</tr>
</tbody>
</table>

Start the Jolt Relay Adapter (JRAD).

1. Type: `tmloadcf -y <UBBFILE>`
2. Type: `tmboot`

Configure the JRAD

A single JRAD process can only be connected to a single JRLY. A JRAD can be configured to communicate with only one JSL and its associated JSH. However, multiple JRADs can be configured to communicate with one JSL. The `CLOPT` parameter for BEA Tuxedo services must be included in the `UBBCONFIG` file.

1. Type: `-l <hexadecimal format>` (The port to the JSL for the JRLY to connect on behalf of the client.)
2. Type: `-c <hexadecimal format>` (The address of the corresponding JSL to which JRAD connects.)
Note: The format is 0x0002PPPN or, in dot notation, 100.100.10.100.

3. Configure networked components. **Jolt is now configured.**

### Jolt Background Information

This section contains additional information on Jolt components.

### Jolt Server

The Jolt Server is a listener that supports one or more handlers.

**Jolt Server Listener (JSL).** The JSL is configured to support clients on an IP/port combination. The JSL works with the Jolt Server Handler (JSH) to provide client connectivity to the backend of the Jolt system. The JSL runs as a BEA Tuxedo server.

**Jolt Server Handler (JSH).** The JSH is a program that runs on a BEA Tuxedo server machine to provide a network connection point for remote clients. The JSH works with the JSL to provide client connectivity residing on the backend of the Jolt system. More than one JSH can be available to the JSL, up to 32,767. (See description of the `-M` command-line option on page 3-26.)

**System Administrator Responsibilities.** The system administrator’s responsibilities for the server components of Jolt include:

- Determining the JSL network address.
- Determining the number of Jolt clients to be serviced. (The number of clients to be serviced is limited by `MAXWSCLIENTS` in UBB.)
- Determining the minimum and maximum number of JSHs.

### Starting the JSL

To start all administrative and server processes in the UBBCONFIG file:
1. Type: `tmloadcf`
   This command parses the configuration file and loads the binary version of the configuration file.

2. Type: `tmboot -y`
   This command activates the application specified in the configuration file.

   If you do not enter any options, a prompt asks you if you really want to overwrite your TUXCONFIG file.

   See the *Administering the BEA Tuxedo System* or the *BEA Tuxedo Reference Manual* for information on `tmloadcf` and `tmboot`.

### Shutting Down the JSL

All shutdown requests to the Jolt servers are initiated by the BEA Tuxedo command:

```
tmshutdown -y.
```

During shutdown:
- No new client connections are accepted.
- All current client connections are terminated. BEA Tuxedo rolls back in-flight transactions. Each client receives an error message indicating that the service is unavailable.

### Restarting the JSL

BEA Tuxedo monitors the JSL and restarts it in the event of a failure. When BEA Tuxedo restarts the listener process, the following occurs:

- Clients attempting a listener connection must try to reconnect. Clients attempting a handler connection receive a timeout or a time delay.
- Clients currently connected to a handler are disconnected (JSH exits when its corresponding JSL exits normally).
Configuring the JSL

The Jolt Server Listener (JSL) is a BEA Tuxedo server responsible for distributing connection requests from Jolt to the Jolt Server Handler (JSH). BEA Tuxedo must be running on the host machine where the JSL and JREPSVR are located.

JSL Command-Line Options

The server may need to obtain information from the command line. The CLOPT parameter allows you to specify command-line options that can change some defaults in the server. The JSL command-line options are explained in Table 16-2.

<table>
<thead>
<tr>
<th>JSL Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-c compression_threshold]</td>
<td>Enables application data sent between a Jolt client and a Jolt server (JSH) to be compressed during transmission over the network.</td>
</tr>
<tr>
<td></td>
<td>compression_threshold is a number that you specify between 0 and 2,147,483,647 bytes. Any messages that are larger than the specified compression threshold are compressed before transmission.</td>
</tr>
<tr>
<td></td>
<td>The default is no compression; that is, if no compression threshold is specified, Jolt does not compress messages on client or server.</td>
</tr>
<tr>
<td></td>
<td><strong>Note:</strong> The previous (-c connection-mode) option has been replaced with the (-j connection-mode) option.</td>
</tr>
<tr>
<td>[-d device_name]</td>
<td>The device for platforms using the Transport Layer Interface. There is no default. Required. (Optional for sockets)</td>
</tr>
</tbody>
</table>
**Table 16-2 JSL Command-Line Options**

<table>
<thead>
<tr>
<th>JSL Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-H external netaddr</code></td>
<td><code>external netaddr</code> is the network address Jolt clients use to connect to the application. The JSL process uses this address to listen for clients attempting to connect at this address. If the address is <code>0x0002MMMMdddddddd</code> and JSH network address is <code>0x00021111ffffffff</code>, the known network address is <code>0x00021111dddd dddd</code>. If the address starts with <code>/</code> network address, the type is IP based and the TCP/IP port number of JSH network address is copied into the address to form the combined network address. The IP address must be specified in the following form: <code>-H //external ip address:M MMMM</code> (Optional for JSL in BEA Tuxedo 6.4 and 6.5)</td>
</tr>
<tr>
<td><code>-I init-timeout</code></td>
<td>The time (in seconds) that a Jolt client is allowed to complete initialization through the JSH before it is timed out by the JSL. Default is 60 seconds. (Optional)</td>
</tr>
<tr>
<td><code>-j connection_mode</code></td>
<td>The following connection modes from clients are allowed: RETAINED - the network connection is retained for the full duration of a session. RECONNECT - the client establishes and brings down a connection when an idle timeout is reached, reconnecting for multiple requests within a session. ANY - the server allows a client to request either a RETAINED or RECONNECT type of connection for a session. The default is ANY. That is, if no option is specified, the server allows a client to request either a RETAINED or RECONNECT type of connection. (Optional) Note: This option has been changed in this release from <code>-c [connection_mode]</code> to <code>-j [connection_mode]</code>.</td>
</tr>
<tr>
<td><code>-m minh</code></td>
<td>The minimum number of JSHs that are available in conjunction with the JSL at one time. The range of this parameter is from 0 through 255. Default is 0. (Optional)</td>
</tr>
</tbody>
</table>
Configuring the BEA Jolt System

Table 16-2  JSL Command-Line Options

<table>
<thead>
<tr>
<th>JSL Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-H external netaddr]</td>
<td>external netaddr is the network address Jolt clients use to connect to the application. The JSL process uses this address to listen for clients attempting to connect at this address. If the address is 0x0002MMMMddddd and JSH network address is 0x00021111ffffffff, the known network address is 0x00021111dddd dddd. If the address starts with &quot;//&quot; network address, the type is IP based and the TCP/IP port number of JSH network address is copied into the address to form the combined network address. The IP address must be specified in the following form: <code>-H //external ip address:MMMM</code> (Optional for JSL in BEA Tuxedo 6.4 and 6.5)</td>
</tr>
<tr>
<td>[-I init-timeout]</td>
<td>The time (in seconds) that a Jolt client is allowed to complete initialization through the JSH before it is timed out by the JSL. Default is 60 seconds. (Optional)</td>
</tr>
<tr>
<td>[-j connection_mode]</td>
<td>The following connection modes from clients are allowed: RETAINED - the network connection is retained for the full duration of a session. RECONNECT - the client establishes and brings down a connection when an idle timeout is reached, reconnecting for multiple requests within a session. ANY - the server allows a client to request either a RETAINED or RECONNECT type of connection for a session. The default is ANY. That is, if no option is specified, the server allows a client to request either a RETAINED or RECONNECT type of connection. (Optional)</td>
</tr>
<tr>
<td>[-m minh]</td>
<td>The minimum number of JSHs that are available in conjunction with the JSL at one time. The range of this parameter is from 0 through 255. Default is 0. (Optional)</td>
</tr>
</tbody>
</table>

Note: This option has been changed in this release from `-c [connection_mode]` to `-j [connection_mode].`
**Jolt Background Information**

### Table 16-2 JSL Command-Line Options

<table>
<thead>
<tr>
<th>JSL Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-M maxh]</td>
<td>The maximum number of JSHs that are available in conjunction with the JSL at one time. If this option is not specified, the parameter defaults to MAXWSCLIENTS divided by the rounded-up ( \times ) multiplexing factor (MPX). If specified, the (-M) option takes a value from 1 to 32,767. (Optional)</td>
</tr>
</tbody>
</table>
| [-n netaddr]            | Network address used by the Jolt listener with BEA Tuxedo 6.4 and 6.5, and WebLogic Enterprise 4.2. TCP/IP addresses may be specified in the following formats: 
"//host.name:port_number"
"#/.#.#.#:port_number"
In the first format, the domain finds an address for hostname using the local name resolution facilities (usually DNS). Hostname must be the local machine, and the local name resolution facilities must unambiguously resolve hostname to the address of the local machine.
In the second example, the “#/.#.#.” is in dotted decimal format. In dotted decimal format, each # should be a number from 0 to 255. This dotted decimal number represents the IP address of the local machine. |
Configuring the BEA Jolt System

Table 16-2 JSL Command-Line Options

<table>
<thead>
<tr>
<th>JSL Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>-T Client-timeout</code></td>
<td>The time (in minutes) allowed for a client to stay idle. If a client does not make any requests during this time, the JSH disconnects the client and the session is terminated. If an argument is not supplied, the session does not timeout. When the <code>-j ANY</code> or <code>-j RECONNECT</code> option is used, always specify <code>-T</code> with an idle timeout value. If <code>-T</code> is not specified and the connection is suspended, JSH does not automatically terminate the session. The session never terminates if a client abnormally ends the session. If a parameter is not specified, the default is no timeout. (Optional)</td>
</tr>
</tbody>
</table>

| `-w JSH` | This command line option indicates the Jolt Server Handler. Default is JSH. (Optional) |

| `-x mpx-factor` | This is the number of clients that one JSH can service. Use this parameter to control the degree of multiplexing within each JSH process. If specified, this parameter takes a value from 1 to 32767 for UNIX and NT. Default value is 10. (Optional) |
Security and Encryption

Authentication and key exchange data are transmitted between Jolt clients and the JSL/JSH using the Diffie-Hellman key exchange. All subsequent exchanges are encrypted using RC4 encryption. International packages use a DES key exchange and a 128-bit key, with 40 bits encrypted and 88 bits exposed.

Programs using the 128-bit encryption cannot be exported outside the United States without proper approval from the United States government. Customers with intranets extending beyond the United States cannot use this mode of encryption if any internal clients are outside the United States.

Jolt Relay

The combination of the Jolt Relay (JRLY) and its associated Jolt Relay Adapter (JRAD) is typically referred to as the Internet Relay. Jolt Relay is a component that routes messages from a Jolt client to a JSL or JSH. This eliminates the need for the JSH and BEA Tuxedo to run on the same machine as the Web server (generally considered as insecure). The Jolt Relay consists of two components illustrated in Figure 16-2.

- Jolt Relay (JRLY). The JRLY is the Jolt Relay front-end. It is not a BEA Tuxedo client or server and is not dependent on the BEA Tuxedo version. It is a stand-alone software component. It requires only minimal configuration to allow it to work with Jolt clients.

<table>
<thead>
<tr>
<th>JSL Command Line Option</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[-2 0</td>
<td>40</td>
</tr>
</tbody>
</table>
**Jolt Relay Adapter (JRAD).** The JRAD is the Jolt Relay back-end. It is a BEA Tuxedo system server, but does not include any BEA Tuxedo services. It requires command line arguments to allow it to work with the JSL and the BEA Tuxedo system.

**Note:** The Jolt Relay is transparent to Jolt clients and Jolt servers. A Jolt server can simultaneously connect to intranet clients directly, or via the Jolt Relay to Internet clients.

**Figure 16-2  Jolt Internet Relay Path**

![Diagram of Jolt Internet Relay Path](image)

Figure 16-2 shows how a browser connects to the Web server software and downloads the Jolt applets. The Jolt applet or client connects to the JRLY on the Web server machine. The JRLY forwards the Jolt messages across the firewall to the JRAD. The JRAD selectively forwards messages to the JSL or appropriate JSH.
Jolt Relay Failover

There are two points of failovers associated with JRLY:

- Jolt Client to JRLY connection failover
- JRLY to JRAD connection failover

Jolt Client to JRLY Connection Failover

If one server address does not result in a successful session, the failover function allows the Jolt Client API to connect to the next free (unconnected) JRLY specified in the argument list of the API. To enable this failover in an NT environment, multiple NT JRLY services can be executed. In a non-NT environment, multiple JRLY processes are executed. Each JRLY (service or process) has its own configuration file. This type of failover is handled by Jolt 1.2 client API changes that allow you to specify a list of Jolt server addresses (JSL or JRLY).

JRLY to JRAD Adapter Connection Failover

Each JRLY configuration file has a list of JRAD addresses. When a JRAD is unavailable, JRLY tries to connect to the next free (unconnected) JRAD, in a round-robin fashion. Two JRLYs cannot connect to the same JRAD. Given these facts, you can make the connection efficient by giving different JRAD address orders. That is, if you make one extra JRAD available on standby, the first JRLY that loses its JRAD connects to the extra JRAD. This type of failover is handled by JRLY alone.

If any of the listed JRADs are not executing when JRLY is started, the initial connection fails. When a Jolt client tries to connect to JRLY, the JRLY again tries to connect to the JRAD.

To accommodate the failover functionality, you have to boot multiple JRADs by configuring them in the UBBCONFIG file.
Jolt Relay Process

The JRLY (front-end relay) process can be started before or after the JRAD is started. If the JRAD is not available when the JRLY is started, the JRLY attempts to connect to the JRAD when it receives a client request. If JRLY is still unable to connect to the JRAD, the client is denied access and a warning is written to the JRLY error log file.

Starting the JRLY on UNIX

Start the JRLY process by typing the command name at a system prompt.

```
jrly -f <config_file_path>
```

If the configuration file does not exist or cannot be opened, the JRLY prints an error message. Refer to Appendix B of the BEA Jolt Developer's Guide for the Jolt Relay error messages.

If the JRLY is unable to start, it writes a message to standard error and attempts to log the startup failure in the error log, then exits.

JRLY Command-Line Options for NT

This section discusses command-line options that are available from the NT version of JRLY.exe. Note that:

- JRLY as an NT service is available only for Microsoft Windows NT.
- When the display suffix is optional (when `display_suffix` is shown), all operations are performed on the default JRLY NT service instance.
- For manually-installed, additional JRLY services, a suffix (any string) is required. Also, you can install the default service manually by omitting the optional string suffix.
- Each instance of JRLY NT service uses the same binary executable file.
- A new process is started for each instance of JRLY NT service.
- The syntax for these options is: `jrly -command`.
- Text specified within brackets ([ ]) is optional.
All the following commands except \texttt{-start} and \texttt{-stop} require that you have write access to Windows NT Registry.

The \texttt{-start} and \texttt{-stop} commands require that you have NT Service control access. These restrictions are based on NT user restrictions.

The command line options are as follows:

\texttt{jrly -install \{display\_suffix\}}

Install \texttt{jrly} as an NT service.

\textbf{Example 1:} \texttt{jrly -install}

In this example, the default JRLY is installed as an NT Service and is displayed in the Service Control Manager (SCM) as \textbf{Jolt Relay}.

\textbf{Example 2:} \texttt{jrly -install MASTER}

In this case, an instance of JRLY is installed as an NT Service and is displayed in the SCM as \textbf{Jolt Relay\_MASTER}. The suffix, MASTER, does not have any significance; it is only used to uniquely identify various instances of JRLYs.

\textbf{Discussion:} At this point, this instance of JRLY is not ready for starting. It must be assigned the configuration file (see the \texttt{set} command discussion) that specifies the listening TCP/IP port, JSH connection TCP/IP port, log files, and sockettimeout. This file should not be shared between various instances of JRLY.

\texttt{jrly -remove \{display\_suffix\} | -all}

Remove one or all JRLY from NT service.

If \texttt{\{display\_suffix\}} is specified, this command removes the specified JRLY service.

If \texttt{\{display\_suffix\}} is not specified, this command removes the default JRLY from being an NT Service.

If the \texttt{-all} option is specified, all the JRLY NT Services are removed. Related NT registry entries under:

\texttt{HKEY\_LOCAL\_MACHINE\System\Current\ControlSet\Services}

and

\texttt{HKEY\_LOCAL\_MACHINE\Software\BEA\Jolt\1.2}
are removed.

jrly -set [-d display_suffix] -f config_file

Update the registry with the full path of a new configuration file.

Example 1:  jrly -set -f c:\tux64\udataobj\jolt\jrly.con

In this example, the default JRLY NT Service (Jolt Relay) is assigned a configuration file called jrly.con that is located in: c:\tux64\udataobj\jolt directory.

Example 2:  jrly -set -d MASTER -f c:\tux64\udataobj\jolt\master.con

Here, the JRLY NT Service instance, called Jolt Relay_MASTER is assigned a configuration file called jrly_master.con that is located in c:\tux64\udataobj\jolt directory.

jrly -manual [display_suffix]

Set the start/stop to manual.

This command sets the specified JRLY instance to be manually controlled, using either the command-line options or the SCM.

jrly -auto [display_suffix]

Set the start/stop to automatic.

This command sets all the operations for specified NT Service to be automatically started when the OS boots and stopped when the OS shuts down.

jrly -start [display_suffix]

Start the JRLY.

This command starts the specified JRLY.

jrly -stop [display_suffix]

Stop the JRLY.

This command stops the specified JRLY.

jrly -version

Print the current version of JRLY.
This command prints the current version of JRLY binary.

```
jrly -help
```

Print command-line options.

This command prints the command-line options with brief descriptions.

## JRLY Command-Line Option for UNIX

There is only one JRLY command-line option for UNIX:

```
jrly -f <config_file_path>
```

Start the JRLY process.

This option starts the JRLY process. If the configuration file does not exist or cannot be opened, the JRLY prints an error message. If the JRLY is unable to start, it writes a message to standard error, attempts to log the startup failure in the error log, then exits.

## JRLY Configuration File

The format of the configuration file is a TAG=VALUE format. Blank lines or lines starting with a “#” are ignored. Refer to Listing 16-3 for an example of the formal specifications of the configuration file.

### Listing 16-3 Specification of Configuration File

```
LOGDIR=<LOG_DIRECTORY_PATH>
ACCESS_LOG=<ACCESS_FILE_NAME in LOGDIR>
ERROR_LOG=<ERROR_FILE_NAME in LOGDIR>
LISTEN=<IP:Port combination where JRLY will accept connections>
CONNECT=<IP:Port combination associated with JRAD>
SOCKETTIMEOUT=<Seconds for socket accept() function>
```
Note: SOCKETTIMEOUT is the duration (in seconds) for which the relay NT service blocks for network activity (new connections, data to be read, closed connections). It is valid only on NT machines.

SOCKETTIMEOUT also affects the SCM. When the SCM requests the service to stop, the SCM needs to wait at least SOCKETTIMEOUT seconds before doing so.

Listing 16-4 shows an example of the JRLY configuration file. The CONNECT line specifies the IP address and port number of JRAD machine.

Listing 16-4   Example of JRLY Configuration File

LOGDIR=/usr/log/relay
ACCESS_LOG=access_log
ERROR_LOG=errorlog
# jrly will listen on port 4444
LISTEN=200.100.10.100:4444
CONNECT=200.100.20.200:4444, 200.100.20.200:5555,...
SOCKETTIMEOUT=30            //See note under Listing 3-4

The format for directory and file names is determined by the operating system. UNIX systems use the forward slash (/). NT systems use the backslash (\). If any of the files specified in LOGDIR, ACCESS_LOG or ERROR_LOG cannot be opened for writing, the JRLY prints an error message on stderr and exits.

The format for host names and port numbers are shown in Table 16-3.

Table 16-3 Host Name and Port Number Formats

<table>
<thead>
<tr>
<th>Host Name/Port Number</th>
<th>Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hostname:Port</td>
<td>Hostname is a string, Port is a decimal number</td>
</tr>
<tr>
<td>//Hostname:Port</td>
<td>Hostname is a string, Port is a decimal number</td>
</tr>
<tr>
<td>IP:Port</td>
<td>IP is a dotted notation IP address, Port is a decimal number</td>
</tr>
</tbody>
</table>
Jolt Relay Adapter

The Jolt Relay Adapter (back-end relay) is a BEA Tuxedo system server. The Jolt Relay Adapter (JRAD) server may or may not be located on the same BEA Tuxedo host machine (in SHM mode) and server group to which the JSL server is connected.

The JRAD can be started independently of its associated JRLY. JRAD tracks its startup and shutdown activity in the BEA Tuxedo log file.

JRAD Configuration

A single JRAD process can only be connected to a single JRLY. A JRAD can be configured to communicate with only one JSL and its associated JSHs. However, multiple JRADs can be configured to communicate with one JSL. The CLOPT parameter for the BEA Tuxedo servers must be included in the UBBCONFIG file. For additional information about the CLOPT parameters, refer to Table 16-4.

Table 16-4  JRAD CLOPT Parameter Descriptions

<table>
<thead>
<tr>
<th>CLOPT Parameter</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>-l &lt;hexadecimal format&gt;</td>
<td>Port to listen for the JRLY to connect on behalf of the client.</td>
</tr>
<tr>
<td>-c &lt;hexadecimal format&gt;</td>
<td>The address of the corresponding JSL to which JRAD connects.</td>
</tr>
<tr>
<td>-H &lt;hexadecimal format&gt;</td>
<td>Used when there is a network address translation performed for JRLY listen address.</td>
</tr>
</tbody>
</table>

Note: The format is 0x0002PPPPNNN. Refer to the BEA Jolt 1.2 Release Notes for additional information on JRAD.
Listing 16-5 shows the sample UBBCONFIG file.

Listing 16-5  Sample JRAD Entry in UBBCONFIG File

```
# JRAD host 200.100.100.10 listens at port 2000, connects to JSL port 8000 on the same host
JRAD    SRVGRP=JSLGRP   SRVID=60
       CLOPT="-A -- -l 0x000207D0C864640A -c 0x00021f40C864640A"
```

Network Address Configurations

There are several networked components that must be configured to work together when configuring a Jolt Internet Relay. Prior to configuration, review the criteria required in Table 16-5 and record the information. This will help minimize the possibility of misconfiguration.

Table 16-5  Jolt Internet Relay Network Address Configuration Criteria

<table>
<thead>
<tr>
<th>JRLY</th>
<th>JRAD</th>
<th>JSL</th>
</tr>
</thead>
<tbody>
<tr>
<td>LISTEN: &lt;Location where the clients connect&gt;</td>
<td>-l: &lt;Location of where the listener connects the JRLY&gt;</td>
<td>-n: &lt;Location of JSL. Must match -c parameter of JRAD&gt;</td>
</tr>
<tr>
<td>CONNECT: &lt;Location of your JRAD. Must match the -l parameter of JRAD&gt;</td>
<td>-c: &lt;Location of JSL. Must match -n parameter of JSL&gt;</td>
<td></td>
</tr>
</tbody>
</table>

Jolt Repository

The Jolt Repository contains BEA Tuxedo service definitions that allow Jolt clients to access BEA Tuxedo services. The Jolt Repository files included with the installation contain service definitions used internally by Jolt. See the *BEA Jolt Developer’s Guide* for detailed instructions on how to add definitions to the application services.
Configuring the Jolt Repository

To configure the Jolt Repository, modify the application UBBCONFIG file. The UBBCONFIG file is an ASCII version of the BEA Tuxedo configuration file. Create a new UBBCONFIG file for each application. See the BEA BEA Tuxedo Reference Manual for information regarding the syntax of the entries for the file. Listing 16-6 shows relevant portions of the UBBCONFIG file.

Listing 16-6   Sample UBBCONFIG File

```
*GROUPS
JREPGRP       GRPNO=94  LMID=SITE1
*SERVERS
JREPSVR SRVGRP=JREPGRP SRVID=98
  RESTART=Y  GRACE=0  CLOPT="-A -- -W -P /app/jrepository"
JREPSVR SRVGRP=JREPGRP SRVID=97
  RESTART=Y  RQADDR=JREPQ  GRACE=0  CLOPT="-A -- -P /app/jrepository"
JREPSVR SRVGRP=JREPGRP SRVID=96
  RESTART=Y  RQADDR=JREPQ  REPLYQ=Y  GRACE=0  CLOPT="-A -- -P /app/jrepository"

Note:   For UNIX systems, use the slash (/) when setting the path to the jrepository file (for example, app/repository). For NT systems, use the backslash (\) and specify the drive name (for example, c:\app\repository).

Change the sections of the UBBCONFIG file indicated in Table 16-6:

Table 16-6   UBBCONFIG File

<table>
<thead>
<tr>
<th>Section</th>
<th>Parameters to be specified</th>
</tr>
</thead>
<tbody>
<tr>
<td>GROUPS</td>
<td>LMID, GRPNO</td>
</tr>
<tr>
<td>SERVERS</td>
<td>SRVGRP, SRVID</td>
</tr>
</tbody>
</table>
GROUPS Section

A GROUPS entry is required for the group that includes the Jolt Repository. The group name parameter is a name selected by the application.

1. Specify the same identifiers given as the value of the LMID parameter in the MACHINES section.

2. Specify the value of the GRPNO between 1 and 30,000 in the GROUPS section.

SERVERS Section

The Jolt Repository server, JREPSVR, contains services for accessing and editing the Repository. Multiple JREPSVR instances share repository information through a shared file. Include JREPSVR in the SERVERS section of the UBBCONFIG file.

1. Indicate a new server identification (for example, 98) with the SRVID parameter.

2. Specify the -W flag for one JREPSVR to ensure that you can edit the Repository. The Repository is read-only without this flag.
   
   **Note:** You must install only one writable JREPSVR (that is, only one JREPSVR with the -W flag). Multiple read-only JREPSVRs can be installed on the same host.

3. Type the -P flag to specify the path of the repository file. An error message displays in the BEA Tuxedo ULOG file if the argument for the -P flag is not entered.

4. Add the file pathname of the Repository file (for example, /app/jrepository).

5. Boot the BEA Tuxedo system using the tmloadcf command (for example, tmloadcf -y ubbconfig) and tmboot command. See the Administering the BEA Tuxedo System for information on tmloadcf and tmboot.

Repository File

A Repository file, jrepository, is available with Jolt. This file includes bankapp services and the Repository services that you can modify, test, and delete using the Repository Editor.

**Note:** The Jolt 1.2 Repository file is different from the Jolt 1.1 Repository file. If you are using Jolt 1.1, you must make applicable changes.
Inside the $jrepository$ file, the service definitions for the services in the Jolt Repository Server (JREPSVR) have FML32 as their buffer types. During installation, the new service definitions for Jolt 1.2 JREPSVR should be appended to the existing $jrepository$ file as part of the upgrade.

Start with the $jrepository$ file provided with the installation, even if you are not going to test the $bankapp$ application with Jolt. Delete the $bankapp$ packages or services that you do not need.

The pathname of the file must match the argument of the `-P` option.

**Warning:** Do not modify the Repository files manually or you will not be able to use the Repository Editor. Although the $jrepository$ file can be modified and read with any text editor, the Jolt system does not have integrity checks to ensure that the file is in the proper format. Any manual changes to the $jrepository$ file might not be detected until runtime. See “Using the Jolt Repository Editor” in the BEA Jolt Developer’s Guide for additional information.

**Initializing Services Using BEA Tuxedo and the Repository Editor**

You must initially define the BEA Tuxedo services using BEA Tuxedo and Jolt in order to make the Jolt services available to the client.

1. Build the BEA Tuxedo server containing the service. See Administering the BEA TUXEDO System or BEA Tuxedo Programmer’s Guide for additional information on the following:
   - Building the BEA Tuxedo applications/server
   - Editing the $UBBCONF$ file
   - Updating the $TUXCONF$ file
   - Administering the $tmboot$ command

2. Access the Jolt Repository Editor. See “Using the Jolt Repository Editor” in the BEA Jolt Developer’s Guide for additional information on the following:
   - Adding a Service
Event Subscription

Jolt Event Subscription is used to receive event notifications from either BEA Tuxedo services or other BEA Tuxedo clients:

**Unsolicited Event Notifications.** These are notifications that a Jolt client receives as a result of a BEA Tuxedo client or service subscribing to unsolicited events, and a BEA Tuxedo client issuing a broadcast (using either a `tpbroadcast()` or a directly targeted message via a `tpnotify()` ATMI call). Unsolicited event notifications do not need the `TMUSREVT` server.

**Brokered Event Notifications.** These notifications are received by a Jolt client via the BEA Tuxedo Event Broker. The notifications are only received when both Jolt clients subscribe to an event and any BEA Tuxedo client or server posts an event using `tppost()`. Brokered event notifications require the `TMUSREVT` server.

Configuring for Event Subscription

Configure the BEA Tuxedo `TMUSREVT` server and modify the application `UBBCONFIG` file. Listing 16-7 shows the relevant sections of `TMUSREVT` parameters in the `UBBCONFIG` file. See the *BEA Tuxedo Programmer’s Guide* for information regarding the syntax of the entries for the file.

**Listing 16-7  UBBCONFIG File**

```
TMUSREVT        SRVGRP=EVBGRP1  SRVID=40        GRACE=3600
ENVFILE="/usr/tuxedo/bankapp/TMUSREVT.ENV"
CLOPT="-e tmusrevt.out -o tmusrevt.out -A --
```
In the SERVERS section of the UBBCONFIG file, modify the SRVGRP and SRVID parameters as needed.

Filtering BEA Tuxedo FML or VIEW Buffers

Filtering is a process that allows you to customize a subscription. If you require additional information about the BEA Tuxedo Event Broker, subscribing to events, or filtering, refer to the BEA Tuxedo Programmer’s Guide, Volume 1.

In order to filter BEA Tuxedo FML or VIEW buffers, the field definition file must be available to BEA Tuxedo at runtime.

Note: There are no special requirements for filtering STRING buffers.

Buffer Types

Table 16-7 shows BEA Tuxedo buffer types:

Table 16-7  BEA Tuxedo Buffer Types

<table>
<thead>
<tr>
<th>Buffer Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FML</td>
<td>Attribute, value pair. Explicit.</td>
</tr>
<tr>
<td>STRING</td>
<td>Length and offset are different values. All readable.</td>
</tr>
<tr>
<td>CARRAY</td>
<td>Character array. Blob of binary data. Only client and server know - JSL doesn’t.</td>
</tr>
<tr>
<td>X_C_TYPE</td>
<td>Equivalent to VIEW.</td>
</tr>
<tr>
<td>X_COMMON</td>
<td>Equivalent to VIEW, but used for both COBOL and C.</td>
</tr>
<tr>
<td>X_OCTET</td>
<td>Equivalent to CARRAY.</td>
</tr>
</tbody>
</table>
FML Buffer Example

Listing 16-8 shows an example using the FML buffer. The FML field definition table is made available to BEA Tuxedo by setting the `FIELDTBLS` and `FLDTBLDIR` variables.

To filter a field found in the `my.flds` file:

1. Copy the `my.flds` file to `/usr/me/bankapp` directory.
2. Add `my.flds` to the `FIELDTBLS` variable in the `TMUSREVT.ENV` file as shown in Listing 16-8:

```
Listing 16-8   FIELDTBLS Variable in the TMUSREVT.ENV File

FIELDTBLS=Usysflds,bank.flds,credit.flds,event.flds,my.flds
FLDTBLDIR=/usr/tuxedo/me/T6.2/udataobj:/usr/me/bankapp
```

If `ENVFILE="/usr/me/bankapp/TMUSREVT.ENV"` is included in the definition of the `UBBCONFIG` file (shown in Listing 16-7), the `FIELDTBLS` and `FLDTBLDIR` definitions are taken from the `TMUSREVT.ENV` file and not from your environment variable settings.

If you remove the `ENVFILE="/usr/me/bankapp/TMUSREVT.ENV"` definition, the `FIELDTBLS` and `FLDTBLDIR` definitions are taken from your environment variable settings. The `FIELDTBLS` and `FLDTBLDIR` definitions must be set to the appropriate value prior to booting the BEA Tuxedo system.

For additional information on event subscriptions and the Jolt Class Library, refer to “Using the Jolt Class Library” in the *BEA Jolt Developer’s Guide*.

**BEA Tuxedo Background Information**

The following sections provide detailed configuration information. Skip this section if you are familiar with BEA Tuxedo.
Configuration File

The BEA Tuxedo configuration file for your application exists in two forms, the ASCII file, UBBCONFIG, and a compiled version called TUXCONFIG. Once you have created a TUXCONFIG, it is best to think of your UBBCONFIG as a backup.

You can make changes to the UBBCONFIG file with your preferred NT editor. Then, at a time when your application is not running, and when you are logged in to your MASTER machine, you can recompile your TUXCONFIG by running tmloadcf(1). System/T prompts you to make sure you really want to overwrite your existing TUXCONFIG file. (If you enter the command with the -y option, the prompt is suppressed.)

The UBBCONFIG File

A binary configuration file called the TUXCONFIG file contains information used by tmboot(1) to start the servers and initialize the bulletin board of a BEA TUXEDO system in an orderly sequence. The binary TUXCONFIG file cannot be created directly. Initially, you must create a UBBCONFIG file. That file is parsed and loaded into the TUXCONFIG using tmloadcf(1). Then tmadmin(1) uses the configuration file or a copy of it in its monitoring activity. tmshutdown(1) references the configuration file for information needed to shut down the application.

Configuration File Format

The UBBCONFIG file can consist of up to nine specification sections. Lines beginning with an asterisk (*) indicate the beginning of a specification section. Each such line contains the name of the section immediately following the *. Allowable section names are: RESOURCES, MACHINES, GROUPS, NETGROUPS, NETWORK, SERVERS, SERVICES, INTERFACES, and ROUTING.

Note: The RESOURCES (if used) and MACHINES sections must be the first two sections, in that order; the GROUPS section must be ahead of SERVERS, SERVICES, and ROUTING.

To configure the JSL, you must modify the UBBCONFIG file. For further information regarding BEA Tuxedo configuration, refer to the BEA Tuxedo Administration Guide. Listing 16-9 shows relevant portions of the UBBCONFIG file.
Listing 16-9  UBBCONFIG File

*MACHINES
MACH1  LMID=SITE1
       MAXWSCLIENTS=40
*GROUPS
JSLGRP  GRPNO=95  LMID=SITE1
*SERVERS
JSL  SRVGRP=JSLGRP  SRVID=30  CLOPT= " -- -n 0x0002PPPPNNNNNNN -d
/dev/tcp -m2 -M4 -x10"

The parameters shown in Table 16-8 are the only parameters that must be designated for the Jolt Server groups and Jolt Servers. You are not required to specify any other parameters.

Change the sections of the UBBCONFIG file shown in Table 16-8.

Table 16-8  UBBCONFIG File Sections

<table>
<thead>
<tr>
<th>Section</th>
<th>Parameters to be specified</th>
</tr>
</thead>
<tbody>
<tr>
<td>MACHINES</td>
<td>MAXWSCLIENTS</td>
</tr>
<tr>
<td>GROUPS</td>
<td>GRPNO, LMID</td>
</tr>
<tr>
<td>SERVERS</td>
<td>SRVGRP, SRVID, CLOPT</td>
</tr>
</tbody>
</table>

MACHINES Section

The MACHINES section specifies the logical names for physical machines for the configuration. It also specifies parameters specific to a given machine. The MACHINES section must contain an entry for each physical processor used by the application. Entries have the form:

ADDRESS or NAME required parameters [optional parameters]

where ADDRESS is the physical name of the processor, for example, the value produced by the UNIX system `uname -n` command.
LIMID=string_value

This parameter specifies that the string_value is to be used in other sections as the symbolic name for ADDRESS. This name cannot contain a comma, and must be 30 characters or less. This parameter is required. There must be an LIMID line for every machine used in a configuration.

MAXWSCLIENTS=number

The MAXWSCLIENTS parameter is required in the MACHINES section of the configuration file. It specifies the number of accessor entries on this processor to be reserved for Jolt and /WS clients only. The value of this parameter must be between 0 and 32,768, inclusive.

The Jolt Server and /WS use MAXWSCLIENTS in the same way. For example, if 200 slots are configured for MAXWSCLIENTS, this number configures BEA Tuxedo for the total number of remote clients used by Jolt and /WS.

Be sure to specify MAXWSCLIENTS in the configuration file. If it is not specified, the default is 0.

Note: If MAXWSCLIENTS is not set, the JSL does not boot.

GROUPS Section

This section provides information about server groups, and must have at least one server group defined in it. A server group entry provides a logical name for a collection of servers and/or services on a machine. The logical name is used as the value of the SRVGRP parameter in the SERVERS section to identify a server as part of this group. SRVGRP is also used in the SERVICES section to identify a particular instance of a service with its occurrences in the group. Other GROUPS parameters associate this group with a specific resource manager instance (for example, the employee database). Lines within the GROUPS section have the form:

GROUPNAME required parameters [optional parameters]

where GROUPNAME specifies the logical name (string_value) of the group. The group name must be unique within all group names in the GROUPS section and LIMID values in the MACHINES section. The group name cannot contain an asterisk(*), comma, or colon, and must be 30 characters or less.

A GROUPS entry is required for the group that includes the Jolt Server Listener (JSL). Make the GROUPS entry as follows:
1. The group name is selected by the application, for example: JSLGRP and JREPGRP.

2. Specify the same identifiers given as the value of the LMID parameter in the MACHINES section.

3. Specify the value of the GRPNO between 1 and 30,000 in the *GROUPS section.

**Note:** Make sure that Resource Managers are *not* assigned as a default value for all groups in the GROUPS section of your UBBCONFIG file. Making Resource Managers the default value assigns a Resource Manager to the JSL and you receive an error during tmboot. In the SERVERS section, default values for RESTART, MAXGEN, etc., are acceptable defaults for the JSL.

**SERVERS Section**

This section provides information on the initial conditions for servers started in the system. The notion of a server as a process that continually runs and waits for a server group’s service requests to process may or may not apply to a particular remote environment. For many environments, the operating system, or perhaps a remote gateway, is the sole dispatcher of services. When either of these is the case, you need only specify SERVICE entry points for remote program entry points, and not SERVER table entries. BEA Tuxedo system gateway servers would advertise and queue remote domain service requests. Host-specific reference pages must indicate whether or not UBBCONFIG server table entries apply in their particular environments, and if so, the corresponding semantics. Lines within the SERVERS section have the form:

```
AOUT required parameters [optional parameters]
```

where AOUT specifies the file (string_value) to be executed by tmboot(1). tmboot executes AOUT on the machine specified for the server group to which the server belongs. tmboot searches for the AOUT file on its target machine, thus, AOUT must exist in a file system on that machine. (Of course, the path to AOUT can include RFS connections to file systems on other machines.) If a relative pathname for a server is given, the search for AOUT is done first in APPDIR, then in TUXDIR/bin, then in /bin, and then in <path>, where <path> is the value of the last PATH= line appearing in the machine environment file, if one exists. The values for APPDIR and TUXDIR are taken from the appropriate machine entry in the TUXCONFIG file.

Clients connect to Jolt applications through the Jolt Server Listener (JSL). Services are accessed through the Jolt Server Handler (JSH). The JSL supports multiple clients and acts as a single point of contact for all the clients to connect to the application at the network address that is specified on the JSL command line. The JSL schedules work...
for handler processes. A handler process acts as a substitute for clients on remote workstations within the administrative domain of the application. The handler uses a multiplexing scheme to support multiple clients on one port concurrently.

The network address specified for the JSL designates a TCP/IP address for both the JSL and any JSH processes associated with that JSL. The port number identified by the network address specifies the port number on which the JSL accepts new client connections. Each JSH associated with the JSL uses consecutive port numbers at the same TCP/IP address. For example, if the initial JSL port number is 8000 and there are a maximum of three JSH processes, the JSH processes use ports 8001, 8002, and 8003.

**Note:** Misconfiguration of the subsequent JSL results in a port number collision.

**Parameters Usable With JSL**

In addition to the parameters specified in the previous sections, the following parameters can be used with the JSL, although you need to understand how doing so would affect your application.

**SVRGRP=string_value**

This parameter specifies the group name for the group in which the server is to run. *string_value* must be the logical name associated with a server group in the *GROUPS* section, and must be 30 characters or less. This association with an entry in the *GROUPS* section means that *AOUT* is executed on the machine with the *LMID* specified for the server group. This association also specifies the *GRPNO* for the server group and parameters to pass when the associated resource manager is opened. All server entries must have a server group parameter specified.

**SRVID=number**

This parameter specifies an *identifier*, an integer between 1 and 3000, inclusive, that identifies this server within its group. This parameter is required on every server entry, even if the group has only one server. If multiple occurrences of servers are desired, do not use consecutive numbers for *SRVIDs*; leave enough room for the system to assign additional *SRVIDs* up to *MAX*.

**Optional Parameters**

The optional parameters of the *SERVERS* section are divided into boot parameters and runtime parameters.
Boot Parameters

Boot parameters are used by `tmboot` when it executes a server. Once running, a server reads its entry from the configuration file to determine its runtime options. The unique server identification number is used to find the right entry. The following are boot parameters.

\[ \text{CLOPT} = \text{string}_\text{value} \]

The `CLOPT` parameter specifies a string of command line options to be passed to `AOUT` when booted. The `servopts(5)` page in the *BEA Tuxedo Reference Manual: Section 5* lists the valid parameters.

Some of the available options apply primarily to servers under development. For example, the `-r` option directs the server to write a record to its standard error file each time a service request begins or ends.

Other command line options may be used to direct to server’s standard out and standard error to specific files, or to start the server so that it initially advertises a limited set of its available services.

The default value for the `CLOPT` parameter is `-A`, which means that the server is started with all available services advertised.

The maximum length of the `CLOPT` parameter value is 256 characters; it must be enclosed in double quotes.

\[ \text{SEQUENCE} = \text{number} \]

This parameter specifies when this server should be booted or shutdown relative to other servers. If `SEQUENCE` is not specified, servers are booted in the order found in the `SERVERS` section (and shut down in the reverse order). If some servers have sequence numbers specified and others do not, all servers with sequence numbers are booted first from low to high sequence number, then all servers without sequence numbers are booted in the order in which they appear in the configuration file. Sequence numbers must be in the range between 1 and 9999. If the same sequence number is assigned to more than one server, `tmboot` may boot those servers in parallel.
MIN=number

The MIN parameter specifies the minimum number of occurrences of the server to boot by tmboot. If an RQADDR is specified, and MIN is greater than 1, the servers form a Multiple Servers Single Queue (MSSQ) set. The identifiers for the servers are SRVID up to (SRVID + (MAX -1)). All occurrences of the server have the same sequence numbers as well as any other server parameters. The value range for MIN is 0 to 1000. If MIN is not specified, the default value is 1.

MAX=number

The MAX parameter sets the maximum number of occurrences of the server to be booted. Initially, tmboot boots MIN servers, and additional servers can be booted up to MAX occurrences using the -i option of tmboot to specify the associated server identifier. The value range for MAX is 0 to 1000. If no value is specified for MAX, the default is the same as for MIN, or 1. Keep in mind that:

- tmboot starts MIN occurrences unless you explicitly call for more with the -i SRVID option of tmboot
- If RQADDR is specified and MIN is greater than one, an MSSQ set is formed
- If MIN is not specified, the default is 1
- If MAX is not specified, the default is MIN
- MAX is especially important for conversational servers because they are spawned automatically as needed

Runtime Parameters

The runtime parameters are used by the server after it has been started by tmboot. As indicated above, tmboot uses the values found in the TUXDIR, APPDIR and ENVFILE parameters for the MACHINES section when booting the server. It also sets the PATH for the server to:

```
"APPDIR:TUXDIR/bin:/bin:<path>"
```

where <path> is the value of the last PATH= line appearing in the ENVFILE file. The following parameters are runtime parameters.

*ENVFILE=string_value*

The `ENVFILE` parameter for a server can be used to add values to the environment established by `tmboot` during initialization of the server. Variables specified in the file named in the `SERVERS ENVFILE` parameter are set after those in the `MACHINES ENVFILE` used by `tmboot`. These files cannot be used to override `TUXDIR`, `APDIR`, `TUXCONFIG`, or `TUSOFFSET`. The best policy is to include in the server’s `ENVFILE` only those variable assignments known to be needed to ensure proper running of the application.

Note that on the server, this file is processed *after* the server starts. Therefore, it cannot be used to set the pathnames used to find executable or dynamically loaded files needed to execute the server. If you need to perform these tasks, use the machine `ENVFILE` instead.

Within `ENVFILE` only lines of the form
```
VARIABLE = string
```
are allowed. `VARIABLE` must start with an underscore or alphabetic character and can contain only underscore or alphanumeric characters. If the server is associated with a server group that can be migrated to a second machine, the `ENVFILE` must be in the same location on both machines.

*CONV={Y | N}*

`CONV` specifies whether or not the server is a conversational server. `CONV` takes a `Y` value if a conversational server is being defined. Connections can only be made to conversational servers, and rpc requests (via `tpcall(3c)` or `tpcall(3c)`) can only be made to non-conversational servers. For a request/response server, you can either set `CONV=N`, which is the default, or omit the parameter.

*RQADDR=string_value*

`RQADDR` assigns a symbolic name to the request queue of this server. MSSQ sets are established by using the same symbolic name for more than one server (or by specifying `MIN` greater than 1). All members of an MSSQ set must offer an identical set of services and must be in the same server group.

If `RQADDR` is not specified, the system assigns a unique key to serve as the queue address for this server. However, `tmadmin` commands that take a queue address as an argument are easier to use if queues are given symbolic names.
RQPERM=number

The **RQPERM** parameter is used to assign UNIX-style permissions to the request queue for this server. The value of *number* can be between 0001 and 0777, inclusive. If no parameter is specified, the permissions value of the bulletin board, as specified by **PERM** in the **RESOURCES** section, is used. If no value is specified there, the default of 0666 is used (this opens your application up to possible use by any login on the system, so consider this carefully).

REPLYQ={ Y | N }

The **REPLYQ** parameter specifies (with a Y or N) whether or not a reply queue, separate from the request queue, should be established for **AOUT**. If Y is specified, the reply queue is created on the same **LMID** as the **AOUT**. In cases where only one server is using the request queue, replies can be picked up from the request queue without causing problems. However, if the server is a member of an MSSQ set and contains services programmed to receive reply messages, **REPLYQ** should be set to Y so that an individual reply queue is created for this server. If that is not done, the reply will be sent to the request queue shared by all servers for the MSSQ set and there is no way of ensuring that it will be picked up by the server that is waiting for it.

It should be standard practice for all member servers of an MSSQ set to specify **REPLYQ=Y** if replies are anticipated. Servers in an MSSQ set are required to have identical offerings of services, so it is reasonable to expect that if one server in the set expects replies, any server in the set can also expect replies.

RPPERM=number

The **RPPERM** parameter is used to assign permissions to the reply queue. *number* is specified in the usual UNIX fashion (for example, 0600); the value can be between 0001 and 0777, inclusive. If **RPPERM** is not specified, the default value 0666 is used.

This parameter is useful only when **REPLYQ=Y**. If requests and replies are read from the same queue, only **RQPERM** is needed; **RPPERM** is ignored.

RESTART={ Y | N }

The **RESTART** parameter takes a Y or N to indicate whether or not **AOUT** is restartable. The default is N. If the server is in a group that can be migrated, **RESTART** must be Y. Note that a server started with a **SIGTERM** signal cannot be restarted; it must be rebooted.
An application’s policy on restarting servers might vary according to whether the server is in production or not. During the test phase of application development it is reasonable to expect that a server might fail repeatedly, but server failures should be rare events once the application has been put into production. You might want to set more stringent parameters for restarting servers once the application is in production.

**Parameters Associated With RESTART**

- **RCMD=string_value**
  
  If AOUT is restartable, this parameter specifies the command that should be executed when AOUT abnormally terminates. The string, up to the first space or tab, must be the name of an executable UNIX file, either a full pathname or relative to APPDIR. (Don’t attempt to set a shell variable at the beginning of the command.) The command name may be optionally followed by command-line arguments. Two additional arguments are appended to the command line: the GRPNO and SRVID associated with the restarting server. *string_value* is executed in parallel with restarting the server.

  The RCMD parameter can be used to specify a command to be executed in parallel with the restarting of the server. The command must be an executable UNIX system file residing in a directory on the server’s PATH. An example of a possible use would be a command that sends a customized message to the userlog to mark the restarting of the server.

- **MAXGEN=number**
  
  If AOUT is restartable, this parameter specifies that it can be restarted at most \((\text{number} - 1)\) times within the period specified by GRACE. The value must be greater than 0 and less than 256. If not specified, the default is 1 (which means that the server can be started once, but not restarted). If the server is to be restartable, MAXGEN must be equal to or greater than 2. RESTART must be Y or MAXGEN is ignored.

- **GRACE=number**
  
  If RESTART is Y, the GRACE parameter specifies the time period (in seconds) during which this server can be restarted, \((\text{MAXGEN} - 1)\) times. The number assigned must be equal to or greater than 0, and less than 2,147,483,648 seconds (or a little more than 68 years). If GRACE is not specified the default is 86,400 seconds (24 hours). Setting GRACE to 0 removes all limitations; the server can be restarted an unlimited number of times.
Entering Parameters

BEA Tuxedo parameters, including RESTART, ROADDR, and REPLYQ, can be used with the JSL. (See Administering the BEA Tuxedo System for additional information regarding runtime parameters.) Enter the following parameters:

1. To identify the SRVGRP parameter, type the previously defined group name value from the GROUPS section.

2. To indicate the SRVID, type a number between 1 and 30,000 that identifies the server within its group.

3. Verify that the syntax for the CLOPT parameter is as follows:

   \[\text{CLOPT}= \text{"-- -n 0x0002PPPPNNNNNNNN -d /dev/tcp -m2 -M4 -x10"}\]

   **Note:** The CLOPT parameters may vary. Refer to Table 16-2 for pertinent command-line information.

4. If necessary, type the optional parameters:

   - Type the SEQUENCE parameter to determine the order that the servers are booted.
   - Specify \textit{Y} to permit release of the RESTART parameter.
   - Type \textit{0} to permit an infinite number of server restarts using the GRACE parameter.

Sample Applications in Jolt Online Resources

You can access sample code that can be modified for use with Jolt through the BEA Jolt product Web page at:


These samples demonstrate and utilize Jolt features and functionality.

Other Web sites with Java-related information include:
Javasoft Home Page (http://www.java.sun.com/)

In addition, the newsgroups in the comp.lang.java hierarchy contain lists of past articles and communications regarding Java, and are a valuable source of archival material.
CHAPTER

17 Post Installation Considerations for BEA Jolt

This chapter discusses post installation procedures, and includes the following sections:

- Installing JRLY After Normal Installation
- Uninstalling Jolt

Installing JRLY After Normal Installation

The following sections give instructions for installing JRLY.

Installing JRLY on UNIX

JRLY is a stand-alone process provided to run on the same machine as the Web server. On UNIX systems, no changes have been made, and JRLY functions as it did in previous versions of Jolt.
Installing JRLY on NT

On NT systems, when you install JRLY as an NT Service, BEA Jolt 1.2 registers the configuration file when NT boots.

On NT, if you do not install JRLY with the normal Jolt installation, but decide you want it later, you can install it through configuration on the command line with the following command:

1. `jrly -install [display_suffix]`    // registers JRLY
2. `jrly -set [-d display_suffix] -f <config file>`   // registers configuration file
3. `jrly -manual [display_suffix]`      // sets start/stop to manual
   or
   `jrly -auto [display_suffix]`       // sets the start/stop to automatic

Uninstalling Jolt

The following sections describe how to uninstall Jolt.

Uninstalling Jolt From NT

To remove Jolt, follow these instructions:

1. From your Control Panel, select Add/Remove Programs.

   The Add/Remove Programs Properties dialog box appears (See Figure 4-1).

2. From the list of programs shown, select “BEA Jolt”.
3. Click on the **Add/Remove** button.

   A confirmation box appears and asks if you really want to remove the selected application.

4. The **Remove Programs from Your Computer** dialog box appears and the program is removed.

   When the program has been removed, the following message appears in the dialog box: “Uninstall Successfully Completed.”

5. Click the **OK** button.
6. Click the **OK** button in the **Add/Remove Programs Properties** dialog box.

   **Jolt has been removed.**

### Uninstalling Jolt From UNIX

Currently, there is no script for uninstalling Jolt from UNIX. You can, however, uninstall Jolt from your UNIX system by removing the applicable files and directories.
Index

A
administrative password 1-17
requirements 1-17
tlisten.pw file 1-17
administrative privileges 3-4, 13-7
APPDIR 4-14, 4-15
appuid 4-20, 4-21
ASP Connectivity for TUXEDO 14-2, 14-3, 14-9

B
BBL 4-20
BEA Administration Console
  CGI directory 1-16
  CGI program 1-16
  Control Panel 4-6
default filenames 1-15
default pathnames 1-15
  entry page 5-5
  file system requirements 1-7
  help files 1-15
  HTML files 1-15
  image files 1-16
  Java files 1-16
  LOGIN window 5-7
  main window 5-8
  servers required 5-3
  starting 5-5
  tuxadm directory 1-16
  Web server 1-15

BEA Administration Control Panel
  Environment Page 4-5
  IPC Resources page 4-10
  Listener page 4-9
  Logging page 4-7
  Machines page 4-4
BEA TUXEDO
  Jolt Repository Editor
    initializing services using 16-29
Bourne shell 4-12
browser requirements 5-3
buffers, filtering 16-31
Bulletin Board Liaison 4-20

C
cdfs 6-15
CGI directory 1-16
classes
  clientclasses 10-5
  client-side 10-5
  permissions 10-9
  server-side 10-5
  user 10-9
  user-written 10-5
  weblogic.policy file 10-9
classpath
  CLASSPATH environment variable 10-5
    -classpath option 10-6
  Java system classpath 10-6
overview 10-4
setting 10-4
setting for NT service 10-16
setting for running under jview 10-7
setting for WebLogic Enterprise Connectivity (WLEC) 10-21
setting in scripts 10-24
setting on command line 10-18
setting to run applications 10-5
setting to run examples 10-5
upgrading 10-5
WebLogic classpath 10-6
weblogic.class.path 10-6
client support 14-6
clientclasses 10-5
Cloudscape
cloudscape.jar 10-6
setting classpath for 10-9
cluster 10-1
command line 10-18
classpath 10-19
examples of starting WebLogic Server 10-20
starting WebLogic Server from 10-17
command-line options 16-12–16-17
Jolt Relay 16-20
Compaq Tru64 UNIX
platform requirements 6-3
Compatibility mode
Solaris systems 3-5
compression 14-1
configuration 15-1, 16-1, 16-28
Event Subscription 16-7, 16-30
Jolt Relay (JRLY) 16-8
Jolt Relay Adapter (JRAD) 16-9, 16-25
Jolt Repository 16-3, 16-27
*GROUPS section 16-28
*SERVERS section 16-28
Jolt Server Listener (JSL) 16-2, 16-12
network address 16-25, 16-26
quick 16-2
Repository File, jrepository 16-28
configuration file 16-33
format 16-33
Jolt Relay 16-23
container managed persistence 10-19
control panel applet 4-2
customer support contact information xvii
D
dbping 10-15
debuggers, using with jdbcKona/Oracle 11-13
Diffie-Hellman (DH) Key Exchange 14-2, 14-3, 16-17
Disk Management Interface 1-10
documentation
Javadoc 14-2
documentation, where to find it xvi
dynamic link library (dll) 14-12
E
encryption 14-3, 16-17
environment variables
JAVA_HOME 2-25
setting on Microsoft Windows NT 2-25
TUXDIR 3-14, 3-16
environment, setting 10-4
error messages
Cannot Install WebLogic Enterprise IPC Helper Service. 2-8, 12-6
Connect Failed 5-8
evaluation license 10-11
Event Viewer window 4-8
External Data Representation 4-22
F
failover
Jolt Client to JRLY connection 16-19
JRLY to JRAD connection 16-19

G
GROUPS section configuration 16-28

H
hardware requirements 7-2
  BEA Administration Console 5-2
  HP-UX 6-12
  Microsoft Windows 95 and 98 6-30
  Microsoft Windows NT (Intel) 6-25
  Solaris 6-40
heap memory size 10-18

I
IDE
  using with jdbcKona/Oracle 11-13
install 8-1
  as Windows NT Service 10-14
  from a zip archive 9-1
  from InstallShield 8-1
hardware requirements 7-2
JDBC drivers - type 2
  checking connections 11-11
  Oracle 11-3
    client libraries 11-6
    debuggers 11-13
    using an IDE 11-13
    using Jview 11-10
software requirements 7-3
UNIX 9-1
installation 2-44, 15-1, 16-1
  before you begin 14-12
  Jolt ASP Connectivity for TUXEDO 14-9
  Jolt Relay 14-6
  Linux 15-11
  NT 15-1
online documentation 15-1
time required 3-4, 13-7
UNIX 15-11
Windows NT 15-1
installation prerequisites
  backing up files 2-7, 3-3
  Microsoft Windows NT 2-8, 3-4
installation requirements
  client 14-7
  disk storage 14-4, 14-7
  Java Developer’s Kit 14-7
  Jolt ASP Connectivity for TUXEDO 14-9
  server 14-4
installation screens
  CGI directory 2-20
  Choose Destination Location 2-17, 2-18
  Enter Information 2-20
  Insert License Disk 2-23
  Install license 2-23
  License File Successfully Installed 2-24
  License Utility 2-27
  Registration Confirmation 2-12, 2-38
  Select Program Folder 2-21, 2-40
  Set APPDIR environment variable 2-40
  Setup 2-9, 2-38, 12-7
  Setup has successfully verified
    installation 2-43
  Setup is verifying installation 2-42
  Software License Agreement 2-9, 2-38,
    12-8
  System Selection 2-12, 2-38
  Tlisten Password 2-23
  Update AUTOEXEC.BAT 2-40
  User Information 2-10, 2-38, 12-8
  WebLogic Enterprise (C++) /WS
    Environment File 2-42
  WebLogic Enterprise (C++) Application
    Directory 2-40
  Welcome 2-9, 2-38, 12-8
  InstallShield
using to install WebLogic Server 8-1
Internationalization 14-2
interoperability
client/server 14-3
release 14-8
Interprocess Communications 1-18, 4-17
IPC 1-18
IPC parameters 1-18
IPC requirements
   calculating 1-25
IPCKEY 4-14

J
jar 9-3
Java 2
   See JDK 1.2 10-20
   setting up Java security manager 10-9
Java Development Kit. See JDK
Java heap memory 10-18
Java Runtime Environment See JRE
Java security manager 10-19
   and JDK 1.2.1 on UNIX 10-11
   setting up 10-9
   weblogic.policy 10-9
Java system classpath
   -classpath option 10-6
   overview 10-4
Javadoc 14-2
JavaSoft 10-2
JDBC drivers - type 4, upgrading 7-3
JDBC drivers, installing type 2 drivers 11-3
JDK
   JavaSoft 10-2
   SunSoft 10-2
JDK (Java Development Kit)
   installing 10-2
   JDK 1.2
   Java Security Manager 10-9
JDK 1.1
   command line example 10-20
   JDK 1.2
   command line example 10-20
   JDK_HOME setting in scripts 10-24
Jolt client class library 14-8
Jolt client requirements 14-8
Jolt Internet Relay 16-17
Jolt Relay (JRLY)
   as NT service 14-2
   command-line options for NT 16-20
   configuration 16-23
   configuration file 16-23
   connection to JRAD 14-2
   failover 16-19
   installation 14-6
   network address configuration 16-25
   starting 16-20
   supported platforms 14-6
Jolt Relay Adapter (JRAD) 16-25
   configuration 16-25
   starting 16-25
Jolt Repository 16-26
   configuring 16-27
   initializing services 16-3
Jolt Repository Editor
   initializing services using 16-29
Jolt server 16-10
   platforms supported 14-4
   shutting down the 16-11
   starting the 16-10
Jolt Server Listener (JSL)
   *MACHINES section 16-34
   *SERVERS section 16-35
   configuration 16-12, 16-36
   optional parameters 16-37
   parameters usable with 16-37
   restarting 16-11
   UBBCONFIG file 16-34
Jolt WAS 14-2
JoltBeans 14-3
JoltSessionAttributes, multiple JSL
   addresses 14-2
JRE (Java Runtime Environment)
  limitations for use 10-3
jrepository 16-28
JRLY See Jolt Relay
JSE Connectivity for TUXEDO 14-2, 14-3
Jview
  See also Microsoft SDK for Java 10-7
  jview
    and class loader 10-7
    and dynamic deployment 10-7
    command line example 10-21
    starting WebLogic Server under 10-7

K
  keys
    license 10-12

L
  LD_LIBRARY_PATH 4-13, 4-20
  lic.txt file 2-24, 3-11
  license
    .class file 10-13
    adding new keys 10-12
    convertLicense utility 10-14
    evaluation 10-11
    installing 10-11
    keys 10-12
      adding 10-12
      overview 10-11
      updating 10-12
      upgrading 10-13
    WeblogicLicense.xml file 10-12
    XML 10-12
  license agreement 15-2
  license disk 2-24
  licensing instructions
    NT 15-17
    UNIX 15-17
  licensing Jolt 15-16
  Linux, installation 15-11
  LLE
    Security Service installation 12-1, 13-1

M
  MACHINES section
    Jolt Server Listener (JSL) 16-34
  makefile 3-15, 3-17
  management
    WebLogic Console 10-22
  Microsoft SDK for Java
    and class loader 10-7
    starting WebLogic Server under 10-7
  modes
    Compatibility for Solaris 3-5
    Standard for Solaris 3-5
  mounting and unmounting a CD
    HP-UX 6-15
    IBM-AIX 6-23
    Solaris 6-44

O
  online documentation
    installation 15-1
  operating system requirements 5-2
  Oracle
    install JDBC driver 11-3

P
  parameters
    associated with RESTART 16-42
    boot 16-38
    optional for JSL 16-37
    runtime 16-39
    TUXEDO 16-43
    usable with JSL 16-37
  password 10-23
  PATH, setting 10-4
Performance Monitor screen 4-12
performance pack
  installing on Solaris 9-4
  library path 9-4
platforms supported 3-2, 6-2
preinstallation checklist 14-12
printing product documentation xvii
product license disk 2-24, 3-12

R
raw disk space
  arranging 1-11
related information xvii
release migration 14-8
removing from Windows NT 2-44
Repository Editor
  before you start 16-4
  exiting the 16-6
  logon 16-5
  starting from Web server 16-4
  starting, using appletviewer 16-4
requirements
  hardware 7-2
  software 7-3

S
sample applications, online resources 16-43
scripts
  setting JDK_HOME 10-24
  startConsole.cmd 10-22
  startConsole.sh 10-22
  starting WebLogic Console with 10-22
  starting WebLogic Server using 10-22
  upgrading 7-3
security 14-3, 16-17
  Java security manager 10-9
security policy
  and JDK 1.2.1 on UNIX 10-11
  security policy file 10-9
  Security Service
    installation on UNIX 12-1, 13-1
    installation on Windows 12-1
  semaphore parameters 1-19
  serverclasses 10-5
  Setting 11-6
  setting
    PATH 10-4
  setting classpath 10-4
  shared memory 1-23
  simpapp 2-28, 2-32, 3-13, 3-15
  simpapp, online resources 16-43
software components
  WebLogic Enterprise Client Only (C++)
  1-5
  WebLogic Enterprise Full System (C++)
  1-5
software installation prerequisites
  backing up files 2-7, 3-3
  Microsoft Windows NT 2-8, 3-4
software installation time
  Microsoft Windows NT 2-8
  UNIX systems 3-4, 13-7
software requirements 7-3
  HP-UX 6-12
  Microsoft Windows 95 and 98 6-30
  Solaris 6-4, 6-19, 6-34, 6-40
Solaris
  performance pack 9-4
Solaris systems
  compatibility mode 3-5
  standard mode 3-5
SSL
  Security Service installation 12-1, 13-1
Standard mode
  Solaris systems 3-5
Start menu
  configuring parameters to run from 10-14
  starting WebLogic Server from 10-14
t3config 10-14
starting WebLogic Server
clusters 10-1
from the command line 10-17, 10-18
classpath 10-19
examples 10-20
JDK 1.1 10-20
JDK 1.2 (Java 2) 10-20
jview 10-21
from the Start Menu 10-14
from the WebLogic Console 10-22
java heap memory 10-18
overview 10-1
starting on boot-up 10-14
using scripts 10-22
WebLogic Enterprise Connectivity 10-21
Windows NT 10-14
from the Start Menu 10-14
NT Service 10-14
selecting a JRE 10-14
t3config 10-14
startWebLogic.cmd 10-24
startWebLogic.sh 10-24
startWebLogicJview.cmd 10-24
superuser privileges 3-4, 13-7
support
  technical xviii
system
PATH 10-4
system requirements 14-3
WebLogic Enterprise 14-4

setting permissions 1-17
TLOG 4-19
TLOGDEVICE 4-19
TLOGSIZE 4-19
tmboot 3-14, 3-16
tmlodecf 1-25, 4-15
tmshutdown 2-44, 3-15, 3-17, 3-19
Tru64 UNIX
  platform requirements 6-3
tuning parameters
  HP-UX 6-16
  IBM-AIX 6-23
  Microsoft Windows NT (Intel) 6-29
  semaphores 1-19
  Solaris 6-23, 6-44
  UNIX systems 1-18
tuxadm
  directory pathname 1-16
TUXCONFIG 4-13, 4-14, 4-20
TUXDIR 3-14, 3-16, 4-12, 4-14
TUXEDO
  background information 16-32
  JSE Connectivity for 14-3
  parameters, entering 16-43
  version 14-3
tuxwsvr 5-3, 5-4, 5-5
tuxwsvr.ini file 5-5

U
UBBCONFIG 1-25, 4-20
  editing 4-13
  example file 4-18
  impact on size of TUXCONFIG file 1-13
  Jolt Server Listener (JSL) configuration
    sample 16-34
  MAXUP parameter 1-24
  NLSADDR parameter 4-21
  relation to TUXCONFIG file 4-14
  SEMMNS parameter 1-19
  TLOGDEVICE parameter 4-19

BEA WebLogic Enterprise Installation Guide   I-7
TYPE parameter 4-22
UBBCONFIG file 16-33
uninstall procedure 2-44
uninstalling
   Jolt (from NT) 17-2
   Jolt (from UNIX) 17-4
Universal Device List 4-19
UNIX
   group 9-2
   install 9-1
   installing on 9-2
   permissions 9-2, 9-4
   running as web server 9-4
   setting library path 9-4
   user 9-2
   weblogic.properties 9-3
UNIX systems
   messages and message queues 1-20
   tuning parameters 1-24
unpack
   jar 9-3
   zip 9-3
upgrading 7-3
   classpath 10-5
   EJB 10-5
   JDBC type 4 drivers 7-3
   license 10-13
   scripts 7-3

V
   version.exe 10-17
   VTOC 1-11

W
   Web servers supported 14-4
   webgui.ini file 5-5, 5-8
   WebLogic classpath 10-6
   WebLogic COM 10-8
   WebLogic Console 10-22

WebLogic Enterprise (WLE) 14-2
WebLogic Enterprise Connectivity 10-21
WebLogic ZAC 9-4
weblogic.class.path 10-6
   overview 10-4
weblogic.policy file 10-9
   and JDK 1.2.1 on UNIX 10-11
   modifying 10-9
   modifying for third party classes 10-10
   modifying for user-written classes 10-10
weblogic.properties file
   modifying 9-3
weblogic.system.home 10-19, 10-21
Windows convenience programs 10-15
   dbping 10-15
   install.exe 10-16
   remove.exe 10-16
   t3config.exe 10-16
   t3console.exe 10-17
   t3server.exe 10-17
   version.exe 10-17
Windows NT
   installation 15-1
   NT Service
      install.exe 10-16
      remove.exe 10-16
      running multiple instances of WebLogic Server as NT services 10-15
      t3config.exe 10-16
      t3console.exe 10-17
      t3server.exe 10-17
      version.exe 10-17
      starting WebLogic Server 10-14
      Windows convenience programs 10-15
      wlisten 1-17, 5-4, 5-5, 5-8

Y
   Y2K 14-2
Z
ZAC 9-4
zip, installing from zip archive 9-3