Content Server Enterprise Edition

Version: 5.5

Installing Content Server with BEA WebLogic Server

Document Revision Date: Oct. 30, 2003

FatWire SOFTWARE

Installing Content Server with BEA WebLogic Server
Document Revision Date: Oct. 30, 2003
Product Version: 5.5

FatWire Technical Support
Web: http://www.fatwire.com/Support

FatWire Headquarters
FatWire, inc.
330 Old Country Road
Suite 207
Mineola, NY 11501
Web: www.fatwire.com

Table of

Contents

Section 1: Overview

Section 2: Database

Section 3: Web Server

Section 4: Application Server

Section 5: Content Server

Section 6: Appendices

Section 1

Overview

This section provides all the information you will need before you begin the installation. It contains the following chapters:

- Chapter 1, “Planning the Installation”
- Chapter 2, “Sequence of Installation”
- Chapter 3, “Worksheets To Document the Installation”

Chapter 1

Planning the Installation

This book describes how to install Content Server with BEA WebLogic Server version 8.1 and version 7.1. If your application server is not BEA WebLogic, refer to one of the following manuals:

- *Installing Content Server with IBM WebSphere*
- *Installing Content Server with* Sun ONE

This chapter is divided into two sections:

- Software Overview
- Hardware Overview

Software Overview

This section describes key areas that you should consider when planning your Content Server installation.

Supported J2EE Components

Content Server is a powerful J2EE application. Installing a J2EE application requires installation expertise with J2EE components such as:

- A web server
- A DBMS
- A JDBC driver
- An application server

Content Server only supports certain versions of these J2EE components. Make sure that you are using the versions listed at the following URL:

`http://cswww.fatwire.com/products/ContentServer`

Who Should Install the Software?

This book is for professionals who have experience installing J2EE components.

If you do not have experience installing J2EE components, **we strongly recommend contacting our Installation Services group** or an experienced Content Server system integrator.

What This Manual Does *Not* Describe

This manual does not contain full installation instructions for the J2EE components. Rather, it explains how to configure the components to work with Content Server. For example, this guide does not explain how to install the Oracle DBMS, but does explain how to configure the Oracle DBMS to work with Content Server.

When you install the components, first refer to the related vendor documentation, and then use the information in this guide as a supplement.

Before Installing the Software

Before installing the J2EE components or Content Server, do the following:

- Call Technical Support and let them know you are ready to install.
- Read the latest *Content Server Release Notes*. They are updated periodically and posted to the FatWire Content Server e-docs web site:

  `http://e-docs.fatwire.com/CSEE/5.5.0/index.htm`
- Make sure that all machines on which you will install J2EE components have a static IP address and a proper DNS configuration.
- If your Content Server system serves content over a secure connection, you must obtain an SSL certificate from a certifying authority such as VeriSign. Obtaining the certificate may take some time, so submit your request well in advance to ensure that you have it before installing Content Server.
- If you will be installing software on UNIX, note the following:
  - The Content Server installation program requires an X-Windows server on the UNIX machine(s) on which you install Content Server. (Nearly all UNIX machines already have an X-Windows server.)
  - You typically install several components while logged in as `root`. So, make sure you know the root password.
  - We recommend that you create a UNIX user named `csuser` on all UNIX systems running the application server. The UID (user ID number) for the `csuser` account must be identical on all the UNIX systems that are running the application server.

After Installing Content Server

After installing Content Server on any machine in your system, you install the Content Server applications. If you are installing a cluster, there are several configuration steps for the content applications that you complete after you have installed the content applications. See *Installing the CS Content Applications* for details.

If you plan to use either of the LDAP or NT authentication plugins, you must configure the plugin after you install Content Server but **before** you install the CS content applications. See *Installing the CS Content Applications* for details.

If you plan to use either of the search engine connectors, you can install the search engine either before or after you install the CS Content Applications. See the Verity or AltaVista release notes, as appropriate.

If you want to install CS-Satellite on remote hosts to increase performance of your delivery system, see *Installing CS-Satellite*.

Hardware Overview

You install Content Server and its J2EE components on three or four different environments:

- Development – the environment where web site developers create the elements that provides the web site's structure
- Management (staging) – the environment where content providers create, edit, and manage content
- Delivery (production) – the environment which delivers the live web site to visitors on the Web
- Testing – an optional environment to test performance or capabilities.

Tiers

Each environment consists of a separate set of hardware. In other words, your site requires at least three sets of hardware. The hardware for an environment might consist of a single machines or many machines. Adding more machines to an environment increases the performance and reliability.

Single-Tier Environments

In a **single-tier** environment, all the J2EE components—including Content Server—are installed on the same machine.

Multi-Tier Environments

In a **multi-tier** environment, different J2EE components are installed on different machines. For example, on a three-tier environment, the web server is installed on one machine, the DBMS on another machine, and the application server and Content Server on a third machine. J2EE components are designed to handle multi-tier environments efficiently.

A multi-tier environment provides better performance than a single-tier environment.

Clustered Environments

In a **clustered** environment, the same J2EE component is installed on multiple machines. Most commercial J2EE components support clustering. For example, you can install most application servers on two, four, or more machines in the environment. (Note that you must install Content Server on every machine on which you install the application server.)

Clustering components increases performance and reliability. Clustering provides failover, which means that when one machine becomes unavailable, other machines take over part of the load. Thus, clustered environments provide more uptime than nonclustered environments.

A WebLogic cluster is a group of servers that work together to provide a more powerful, more reliable application platform than a single server. A cluster appears to its clients as a single server but is in fact a group of servers acting as one. The cluster provides two key features that a single WebLogic host cannot:

- **Scalability** – The capacity of a cluster is not limited to a single machine. New servers can be added to the cluster dynamically to increase capacity. The only limitation on cluster membership is that all servers must be able to communicate using the same IP multicast address. If more hardware is needed, a new server on a new machine can be added.
- **High-availability** – A cluster uses the redundancy of multiple servers to insulate clients from failures. The same service can be provided on multiple servers in the cluster. If one server fails, another can take over. This failover ability increases the availability of the application to clients.

Typical Environments

Determining the optimal hardware configuration, including the number of tiers and the level of clustering, requires considerable expertise. Your FatWire sales engineer or FatWire professional services representative can help you with these decisions. More details about configurations and trade-offs are available in the *CS Architect's Guide*. The following list provides some general guidelines:

- Development environments typically require the least powerful hardware. In fact, many are single-tier environments.
- Most management environments are multi-tier. Management systems that need to be in use 24 hours per day also require clustering. Factors in determining the optimum hardware configuration include the number of content contributors, the complexity of the content, and the frequency of publishing.
- Most large delivery environments are multi-tiered and clustered. Factors in determining the hardware configuration include the amount of content delivered during peak times, the percentage of content that can be cached, and the cost of downtime.

The following figure illustrates a typical management and delivery system:

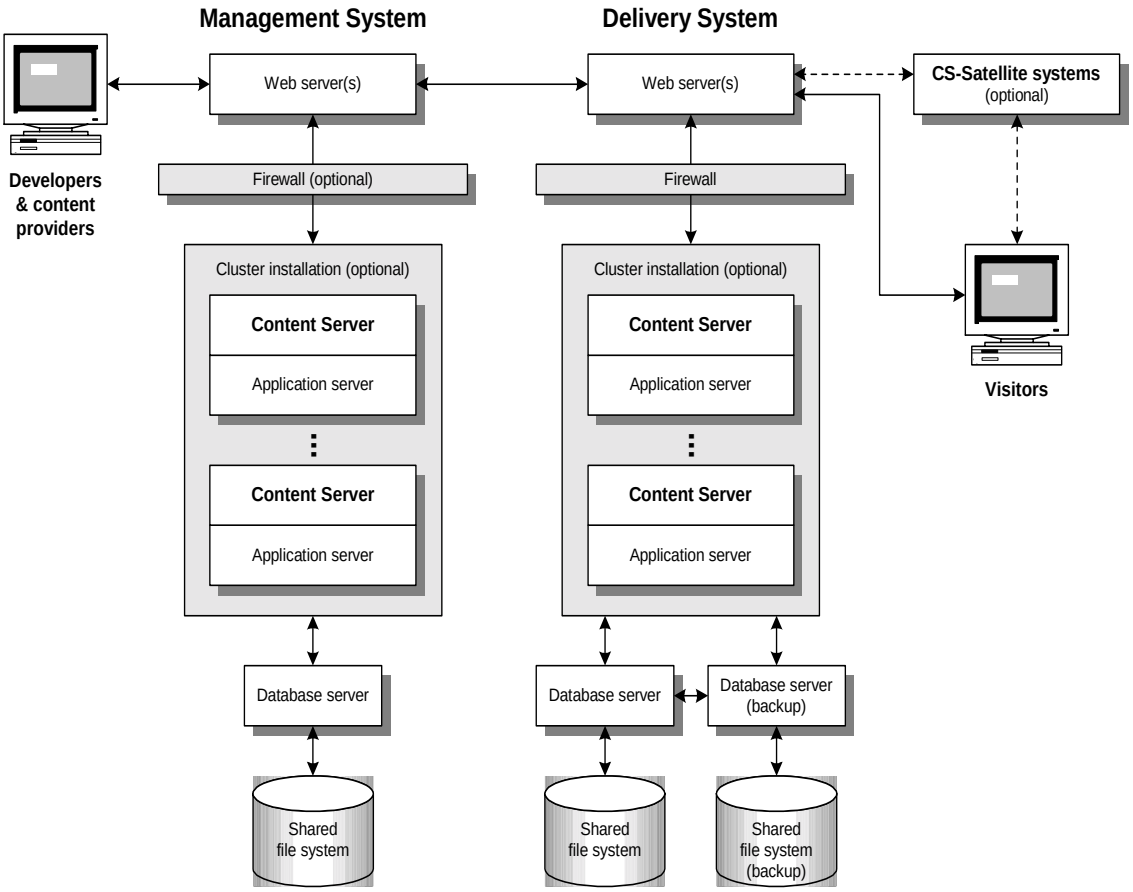

Figure 1: A Typical Management and Delivery Configuration

Minimum Hardware Requirements

Content Server does not impose any minimum hardware requirements. However, the J2EE components on which Content Server runs *do* impose requirements. Because Content Server runs on the same machine as the application server, see the documentation provided by your application server vendor for a list of hardware requirements. Also, see the documentation provided by your web server and DBMS vendors to determine the minimum hardware requirements for those components as well.

Chapter 2

Sequence of Installation

This chapter lists the sequence in which we recommend you install software components. We describe the following:

- The Installation Philosophy
- Additional Optional Items to Install
- Sequence to Install a Multi-Tier Environment
- Sequence to Install a Single-Tier Environment
- Sequence to Install a Clustered Application Server Environment

The Installation Philosophy

The J2EE installation philosophy is this: you install various components and then "connect" them. For example, after installing Oracle and WebLogic, you configure a JDBC driver that allows those two components to communicate.

Content Server is a J2EE web application. Before and after installing Content Server, you must configure WebLogic to handle Content Server, just as you would configure WebLogic to handle any J2EE web application. When you are using WebLogic as your application server, Content Server runs as a WebLogic web application with a WebLogic managed server instance and a WebLogic admin server instance.

In the configuration instructions in this book, we refer to these WebLogic concepts that represent the Content Server application as the Content Server admin server, the Content Server managed server, and the Content Server web application.

The CS content applications become a part of or augment the Content Server web application. That is, when you install the CS content applications, you do not create an additional web application: they become a part of the Content Server web application. Therefore, after you have installed Content Server and the CS content applications, the whole product—including the content applications—is called Content Server.

Additional Optional Items to Install

There are several additional components that you can choose to use with Content Server, including the CS content applications. Two of the three items described in this section, search engines and user authentication plugins, must be installed and configured after you install Content Server but before you install the content applications. The third component, CS-Satellite, must be installed after the content applications are installed.

User Authentication Plugin

If you use LDAP or any other user directory service, configure the appropriate user authentication plugin **before** you install the CS content applications. The installation of CS content applications creates user account records according to the user directory service that is in place at the time of installation.

If you install the CS content applications and then later add a user directory service, you will have to make a large number of adjustments to the existing data.

For information about configuring the LDAP or NT authentication plugins, see *Installing the CS Content Applications*.

CS-Satellite

CS-Satellite is automatically installed with Content Server and this “co-resident” SatelliteServer servlet interacts with the publishing system and the CacheManager servlet. There are no extra steps required to configure the co-resident CS-Satellite although you will most likely want to tune the amount of memory that is allocated to the co-resident CS-Satellite. For more information, see *Installing CS-Satellite*.

You are also encouraged to use the stand-alone version of CS-Satellite on remote web servers to improve the performance of your CS system. If you choose to install CS-Satellite on remote web servers, install them **after** the CS content applications have been installed and tested.

For details, see *Installing CS-Satellite*.

Search Engines

If you have an optional search engine, you can install it either before or after you install the CS content applications.

For information about installing a search engine module, see the AltaVista or Verity release notes, as appropriate for your installation.

Sequence to Install a Single-Tier Environment

The installation sequence for a single-tier system is as follows:

1. Install and verify the DBMS.
2. Install and verify the web server.
3. Install and configure WebLogic.
4. Configure WebLogic and your web server to work together.

5. Configure the JDBC driver to enable the application server to communicate with the database.
6. Configure WebLogic for Content Server.
7. Install, configure, and test Content Server.
8. (Optional) Install, configure, and test a search engine and/or user authentication plugin.
9. Install, configure, and test the CS content applications on the machine that hosts Content Server and WebLogic.

Sequence to Install a Multi-Tier Environment

The installation sequence for a multi-tier system is as follows:

1. Install and verify the DBMS on the database host machine.
2. Install the JDBC driver on the application server host machine.
3. Install and verify the web server on the web server host machine.
4. Install and verify WebLogic on the application server host machine.
5. Configure WebLogic and your web server to work together and then test the configuration.
6. Configure the JDBC driver on your application server host to enable your application server system to communicate with your database.
7. Configure WebLogic for Content Server.
8. Install, configure, and test Content Server on the machine that hosts WebLogic.
9. (Optional) Install, configure, and test a search engine and/or user authentication plugin.
10. Install, configure, and test the CS content applications on the machine that hosts Content Server and WebLogic.

Sequence to Install a Clustered Application Server Environment

The installation sequence for a clustered system can vary greatly depending on networking choices, on whether you also cluster your web servers, and so on, especially, how you choose to configure your WebLogic admin and managed servers. This is the basic sequence of steps during a cluster installation:

1. Install and verify the DBMS on the database host machine. Most likely you also set up a backup database server.
2. Install the JDBC driver on each application server host machine.
3. Install and verify the web server on the web server host machines.

4. Install and verify WebLogic on the primary cluster member, that is, the first application server host machine.
5. Configure WebLogic on the primary cluster member and the web servers to work together and then test the configuration.
6. Configure the JDBC driver on the primary cluster member to enable the WebLogic application server system to communicate with your database.
7. Configure WebLogic for Content Server.
8. Create a shared file system in a location on your network where all the cluster members have access and can both read from and write to.
9. Install, configure, and test Content Server on the primary cluster member.
10. (Optional) Install, configure, and test a search engine and/or user authentication plugin on the primary cluster member.
11. Install, configure, and test the CS content applications on the primary cluster member.
12. Repeat steps 4 through 7 and steps 9 through 11 for each of the remaining cluster member machines.

Chapter 3

Worksheets To Document the Installation

This chapter contains worksheets that list all the parameters that you need to track.

Print this chapter. Then, as you install software, fill in the blank fields in these tables with the values of the specified parameters. You will spare yourself considerable aggravation by doing this. Plus, if something goes wrong during the installation, the information in these worksheets will be valuable while you are troubleshooting. Use a separate set of worksheets for each installation so that each installation is fully documented.

The tables are divided into the following categories:

- DBMS Parameters
- Web Server Parameters
- WebLogic Parameters
- Content Server Parameters

Key to Sample Values

The parameters listed in the tables provide sample values. The sample values fall into the following categories:

- **Default**: a value that is automatically created at the time of the installation.
- **Normal**: a value that represents the normal configuration for a simple installation. Do not use a different value unless your system environment requires it.
- **Option**: a value that you must choose from set list of options.
- **Suggested**: a value that is recommended for the parameter.
- **Example**: an example value that must be replaced by the appropriate one for your installation. The default value probably is not valid on your system.

Note that a **Suggested** account name has an **Example** password value. We strongly recommend that you select a password for this account that is appropriate for the security of your system.

DBMS Parameters

Table 1: DBMS Installation Parameters

Parameter	Shown As	Comments	Your Value
Type and Version	*dbType*	Example: *Oracle 9i*	
Database Host Name	*dbHost*	Example: `centralserve`	
Database Host IP Address	*dbIP*	Example: `101.222.142.173`	
Database Port Number	*dbPort*	Defaults: `1521` (Oracle) `1433` (SQL Server 2000)	
Database Identifier (also called Service Name)	*sid*	Suggested: `csdb`	
Database Root Directory (top-level directory in which DBMS is installed)	*dbroot*	Suggested: `c:\Oracle\`*`oraHome_name`* (Oracle on Windows) `/Oracle/`*`oraHome_name`* (Oracle on Solaris) `c:\Sql2000\`*`sqlHome_name`* (SQL Server on Windows)	

Table 2: DBMS Accounts

Parameter	Shown As	Comments	Your Value
Oracle DBMS Administrator (DBA) Login Name	*dbaname*	Default: `SYSTEM`	
Oracle DBMS Administrator (DBA) Password	*dbapass*	Example: `p055w0rd`	
Content Server Database User Login Name	*csduserbname*	Suggested: `csuserwl`	
Content Server Database User Password	*csdbuserpass*	Example: `c0nt3nt`	

Table 3: Tablespace Parameters

Parameter	Shown As	Comments	Your Value
Default Tablespace or Database Name	*`TblSpace`*	Example: `ContentServer`	
Size of Default Tablespace or Database	*`TblSpaceSize`*	Example: *10 gbytes*	
Temporary Tablespace Name (Oracle)	*`TempSpace`*	Suggested: `TEMP`	
Size of Temporary Tablespace (Oracle)	*`TempSize`*	Example: *2 gbytes*	

Web Server Parameters

Table 4: Web Server Parameters

Parameter	Shown As	Comments	Your Value
Web Version	*`WebVersion`*	Example: Apache 1.3.37	
Web Host Name	*`WebHost`*	Example: `jeeves`	
Web Host IP Address	*`WebIP`*	Example: `104.222.111.155`	
Web Server Port	*`WebPort`*	Default: `80`	
IIS Only: Filter Name (ISAPI plug-in name)	*`FilterName`*	Suggested: `iisforwardfilter`	
Apache Only: Apache Root Directory	*`ApacheRoot`*	Example: `/usr/apache`	

WebLogic Parameters

Table 5: Who Installed WebLogic?

Parameter	Shown As	Comments	Your Value
Installer Account Username	*installerName*	Suggested: `csuser`	
Installer Account Password	*installerPass*	Example: `m1sha`	

Table 6: WebLogic Installation Parameters

Parameter	Shown As	Comments	Your Value
WebLogic Version	*wlVersion*	Example: 7.0 SP2	
WebLogic Host Name	*wlHost*	Example: `jeeves`	
WebLogic Host IP Address (Note: this must be a fixed IP address)	*wlIP*	Example: `101.222.14.17`	
WebLogic Root Directory (BEA Home)	*beaRoot*	Default: `C:\bea` (Windows) `/bea` (Solaris)	
WebLogic Product Directory	*WL_HOME*	Default: `C:\bea\weblogic 81` or `/bea/weblogic81`	
Content Server Domain Name	*csdomain*	Example: `csdomain`	

Table 7: WebLogic Admin Server Parameters

Parameter	Shown As	Comments	Your Value
Admin Server Server Name	*wlAdminSerName*	Example: `csadmin`	
Admin Server Listen Address	*wlAdminHost*	Example: `localhost`	

Parameter	Shown As	Comments	Your Value
Admin Server Listen Port	*wlAdminPort*	Default: `7001`	
Admin Server SSL Port	*wlAdminSSL*	Default: `7002`	
Admin Server Username	*wlAdminName*	Example: `wluser`	
Admin Server Password	*wlAdminPass*	Example: `s3cr3t1v3`	

Table 8: WebLogic Managed Server Parameters

Parameter	Shown As	Comments	Your Value
Managed Server Server Name	*wlManagedSerName*	Suggested: `csmanaged`	
Managed Server Listen Address	*wlManagedHost*	Example: `localhost`	
Managed Server Listen Port	*wlManagedPort*	Default: `8001`	
Managed Server SSL Port	*wlManagedSSL*	Default: `8002`	

Table 9: WebLogic Content Server Parameters

Parameter	Shown As	Comments	Your Value
WebLogic Content Server Name	*wlCSName*	Suggested: `fwcs`	
JNDI Name	*JNDIname*	Suggested: `JNDICS`	

Table 10: WebLogic Cluster Parameters

Parameter	Shown As	Comments	Your Value
Content Server WebLogic Cluster Name	*CSCluster*	Example: `CSCluster`	

Names and IP addresses of all Managed Servers in the Cluster	*CSClusterManagedServer(n)*	Example:
		`csmanageA, 101.222.14.17`
		`csmanageB, 101.123.12.53`	

JDBC Parameters

Table 11: JDBC Parameters

Parameter	Shown As	Comments	Your Value

JDBC Driver Type	*JDBCtype*	Option:
		`* JSQL Connect`
		`* type 2`
		`* type 4`	

JDBC Driver Directory	*JDBCdir*	Suggested: *beaRoot*`/jdbc`	
Net8 Connection String (for Type 2 drivers, Oracle only)	*net8String*	Example: `oraserv1`	
JDBC Connection String	*connString*	See JDBC driver instructions.	
JDBC Connection Pool Name	*poolName*	Suggested: `CSpool`	
Datasource name	*datasource*	Example: `CSDataSource`	

Content Server Parameters

Table 12: Content Server Configuration

Parameter	Shown As	Comments	Your Value
Content Server Version	*csVersion*	Example: 5.0	
Content Server Administrator Username	*csAdminName*	Suggested: `ContentServer`	
Content Server Administrator Password	*csAdminPass*	Example: `c0nt3nt`	
Content Server Root Directory	*csRoot*	Example: `/local/CS`	
Web Server Document Root Directory	*csDocRoot*	Example: `/local/cs/futuretense_cs`	
CS Shared Directory	*csShare*	Accept and record the installation default value.	
Content Server Installation Type	*csType*	Option: **`Single Server`** **`Cluster Member`** **`Upgrade`**	
Satelllite Server Administrator Login	*SatName*	Suggested: `SatelliteServer`	
Satellite Server Administrator Password	*SatPass*	Example: `sputn1k`	

Table 13: Content Server Cluster Parameters

Parameter	Shown As	Comments	Your Value
Shared Directory Name	*`upload`*	Example: `CSshare`	
Sync Folder Name	*`sync`*	Example: `sync`	
`ftsync` value	*`ftsync`*	Suggestion: `CSCluster` (the name of the Content Server cluster name)	

Next Step

Now proceed to the following chapter:

- Chapter 4, “Installing the DBMS”

Section 2

Database

This section describes how to install a database system. It contains the following chapters:

- Chapter 4, “Installing the DBMS”
- Chapter 5, “Configuring the DBMS for Content Server”

Chapter 4

Installing the DBMS

Content Server on WebLogic requires one of the following DBMS:

- Oracle
- SQL Server 2000

Only certain versions of these DBMS are supported; see the following URL for the latest list:

`http://cswww.fatwire.com/products/ContentServer/`

This chapter contains the following sections:

General Considerations

You can install the DBMS on the same machine as the other components or on a separate machine. See the *CSEE Architecture Guide* to determine which is best for you.

Character Sets

The database character set must support all the characters that you intend to store. After you start storing data in the database, it can be tricky to migrate your data to a different character set. Therefore, it is wise to configure your database for the correct character set *before* storing data.

For example, if your database will handle information in European languages only, a database configured for default Latin-1 might suffice. Similarly, if Japanese is the only language used, then the Shift-JIS character set is suitable.

However, if you plan to use the CS-Desktop feature of CS-Direct, you must configure your system to support one of the following character sets, as appropriate:

- Oracle: UTF-8
- SQLServer: Unicode

We recommend that you use the UTF-8 or Unicode character set even if you do not plan to use CS-Desktop. These character sets give you the maximum flexibility. They take up more space in the database, but they encode all characters used in modern languages and in some archaic languages.

Considerations for Installing Oracle

Follow Oracle's instructions for installing the Oracle database server. Content Server imposes no requirements on how you install Oracle.

The easiest way to install Oracle is simply to select the **typical** install option and let the Oracle installation software create an initial database with default settings. Note the following additional guidelines:

- If you are installing Oracle on a single-tier Solaris system that will also host the Apache web server, do **not** have the Oracle installation software install the Apache web server.
- Additionally, in the Database Character Set screen, we recommend that you select **Choose one of the common character sets** and then pick **Unicode standard UTF-8 AL32UTF8**.

Beware

Our customers have run into certain Oracle installation problems in the past. To avoid these problems, we recommend that you consult Oracle's installation documentation, particularly when performing the following tasks:

- Checking the `/etc/system` file
- Creating the Oracle group account
- Creating Oracle user accounts
- Checking environment variables

As you install the DBMS, record information about the installation in Table 1, "DBMS Installation Parameters" on page 22. During the installation, you will create an Oracle database administrator (DBA) account to perform general database administration, such as creating tablespaces or other accounts.

Record the login name and password for the DBA account in Table 2, "DBMS Accounts" on page 22.

Considerations for Installing SQL Server 2000

Follow Microsoft's instructions for installing the SQL Server 2000 database server. Content Server imposes no requirements on how you install SQL Server 2000.

The easiest way to install the database is to select the **typical** install option, and let the SQL Server 2000 installation software create an initial database with its general defaults.

As you install the DBMS, record information about the installation in Table 1, "DBMS Installation Parameters" on page 22.

Next Step

After installing the DBMS, proceed to Chapter 5, "Configuring the DBMS for Content Server."

Chapter 5

Configuring the DBMS for Content Server

This chapter explains how to configure the DBMS for Content Server. It contains the following sections:

- Configure the DBMS
- Validate the Database Configuration
- Install the JDBC Driver on the Application Server Hosts
- Next Step

Use the instructions in the sections that are appropriate for your operating system.

Configure the DBMS

This section presents one set of instructions for configuring SQL Server and one set for configuring Oracle. Complete the procedures in the section that is appropriate for your DBMS.

Configuring SQL Server on Windows

There are three basic steps for configuring SQL Server. They are described in the following sections:

- SQL Server Authentication
- Create the Content Server Database (SQL Server)
- Create the Content Server Database User Account (SQL Server)

SQL Server Authentication

By default, SQL Server is configured to use Windows authentication only. Because Content Server is a web application that will use this database, you must change the authentication mode to SQL Server authentication.

To change the authentication mode to SQL Server authentication, complete the following steps:

1. Invoke the SQL Server Enterprise Manager utility.
2. Expand the servers and select yours.
3. Right-click on your server name and select **Properties** from the right mouse menu.
4. Select the **Security** tab.
5. Under **Authentication**, select the **SQL Server and Windows** option.
6. Restart SQL Server so the change takes effect.

Create the Content Server Database (SQL Server)

Next you create the database for Content Server and reserve enough disk space for it. For help with calculating the amount of disk space to reserve, consult with your database administrator. You can start with these general guidelines:

- Development systems should reserve at least 100 Mb of space.
- Management and delivery systems might need to reserve several Gb of space, depending on what kind of data your site will store.

Record name of the database in the *`TblSpace`* row in Table 3, “Tablespace Parameters” on page 23.

Create the Content Server Database User Account (SQL Server)

After you create the Content Server database, you create a database user account that Content Server will use when it interacts with the database. This user is referred to as the `csdbuser` (not to be confused with the Unix, Linux, or Windows `csuser` account that you use when you are installing WebLogic and Content Server).

In this book, we refer to the login and password of this account as *`csduserbname`* and *`csdbuserpass`*, respectively. Note that, by default, WebLogic requires all passwords to be at least eight characters long; therefore, you should set *`csdbusername`* to have at least eight characters.

Invoke the SQL Server Enterprise Manager and create the `csdbuser` account. This user needs the following **Database Access roles**:

- public
- db_owner
- db_accessadmin
- db_securityadmin
- db_ddladmin
- db_datareader
- db_datawriter

This user does not need any Server roles.

Record the following information in Table 2, “DBMS Accounts” on page 22:

- In the *`csdbusername`* row, record the name of the user.

- In the *csdbuserpass* row, record the password for this user.

Continue to "Validate the Database Configuration" on page 38.

Configuring Oracle on Solaris, Linux, or Windows

There are two basic steps for preparing your Oracle database for the Content Server installation:

- Create the Content Server Tablespaces (Oracle)
- Create the Content Server Database User Account (Oracle)

Create the Content Server Tablespaces (Oracle)

Use the Oracle Enterprise Manager Console to create the default tablespace and the temporary tablespace for the Content Server database. For example:

- Default: 300 MB

  To determine the actual number, you must estimate/calculate the amount of data that you plan to store in the database.

- Temporary: 40 MB

Record the following information in Table 3, "Tablespace Parameters" on page 23:

- In the *TblSpace* row, record the name of the default tablespace.
- In the *TempSpace* row, record the name of the temporary tablespace.

Create the Content Server Database User Account (Oracle)

During the DBMS installation, you created a DBA (Database Administrator) account. You must now create an additional database account that Content Server will use when interacting with the database. This user is referred to as the `csdbuser` (not to be confused with the Unix, Linux, or Windows `csuser` account that you use when you are installing WebLogic and Content Server).

In this book, we refer to the login and password of this account as *csdbusername* and *csdbuserpass*, respectively. Note that, by default, WebLogic requires all passwords to be at least eight characters long; therefore, you should set *csdbusername* to have at least eight characters.

Log in to your DBMS using the DBA account and create the `csdbuser` account. Give this account the `RESOURCE` and `CONNECT` roles only. Do not give this account DBA privileges. Then, record the following information in Table 2, "DBMS Accounts" on page 22:

- In the *csdbusername* row, record the name of the user.
- In the *csdbuserpass* row, record the password for this user.

Continue to "Validate the Database Configuration" on page 38.

Validate the Database Configuration

After you have created the Content Server database and the `csdbuser` account, perform the following test to verify that the csuser account has the correct access to the Content Server database:

1. Access SQL*Plus.
2. Log in to the default tablespace that you just created for the Content Server database as the `csdbuser`.
3. At the SQL prompt, create a simple table. For example:

   ```
   create table authors (au_id char (11) not null, au_lname
   varchar2 (40) not null);
   ```

   The DBMS should create the table.
4. Add a row to your simple table. For example:

   ```
   insert into authors values ('1001', 'Smith');
   ```

   The DBMS should add the row.
5. Now, drop the simple table. For example:

   ```
   drop table authors;
   ```

   The DBMS should remove the table from the database.

Make sure that this test works—that the `csdbuser` can create the table, add a row, and drop the table. If you cannot successfully complete any of these tasks, verify that you can access the database and verify the permissions for the `csdbuser` account before you do anything else.

When this test works, continue with "Install the JDBC Driver on the Application Server Hosts" on page 39.

Install the JDBC Driver on the Application Server Hosts

Content Server and WebLogic communicate with the DBMS via a JDBC driver. The JDBC driver must be physically located on the application server host. Therefore, the next step is to install the appropriate driver on the server(s) that WebLogic and Content Server.

- For SQL Server, you must use the JSQL Connect for SQL Server 2000 driver.
- For Oracle, you can use either the Oracle type 2 (client) driver or the Oracle type 4 (thin) driver. However, **FatWire recommends that you use the type 2 driver**.

To install the JDBC driver for your DBMS, complete the steps in one of the following procedures, as appropriate for your installation:

- "SQL Server Installations: Installing the JSQL Connect JDBC Driver" on page 39
- "Oracle Installations: Installing the Type 2 (Client) JDBC Driver" on page 40
- "Oracle Installations: Installing the Type 4 (Thin) JDBC Driver" on page 40

About the Oracle Type 2 and Type 4 Drivers

The Oracle type 2 driver supports clobs, which means that it allows for larger amounts of text (viturally unlimited) to be stored in the DBMS. Additionally, the type 2 driver works with other Oracle tools to perform database load balancing and failover. Because this driver performs well and support clobs, FatWire recommends that you use the type 2 (client) driver rather than the type 4 driver.

The type 4 (thin) driver has a limit of 2000 characters in files stored in the DBMS. If a file is larger, it is stored referentially in the database but is physically stored in the Content Server file system. The advantage of this driver is that it is easy to set up.

If you choose to use the Type 4 driver you must remember to set the `cc.bigtext` property in `futuretense.ini` to `VARCHAR(2000)`after you install Content Server and before you run the installer for the CS content applications.

SQL Server Installations: Installing the JSQL Connect JDBC Driver

To install the JSQL Connect driver for your SQL Server DBMS, complete the following steps:

1. Download the JDBC driver from NetDirect, which is currently at the following URL:

   `http://www.j-netdirect.com/Downloads.html`

   The JDBC driver is a zip file. Download the zip file to a convenient directory.

   Note that this driver is a licensed product. You receive a temporary license with the download but you must then contact NetDirect and obtain the appropriate license.
2. Unzip the zip file into any directory on the application server host.
3. Record the complete path to and name of this directory in the *`JDBCdir`* row of Table 11, "JDBC Parameters" on page 26.

Complete these steps on each host machine on which you plan to install the WebLogic Server software. If you are installing a cluster, it's a good idea to install the driver in the

same location on each machine—that is, to use the same path name and directory name on each machine.

Oracle Installations: Installing the Type 2 (Client) JDBC Driver

The type 2 driver is installed during the Oracle installation on the database host machine. If you do not intend to install WebLogic on the database host machine, you must install it on each application server host machine.

Note

Do **not** use the type 2 JDBC driver that BEA provides with WebLogic. Get the JDBC driver from your Oracle cd, instead. Note that if you do not complete the steps in Chapter 11, "Configuring the JDBC Driver" after you have installed WebLogic, the BEA driver will get used by default.

To install the type 2 driver:

1. Obtain your Oracle installation cd.
2. On the host machine on which you plan to install the WebLogic Server software, start the Oracle installation program.
3. Select **Oracle Client** from the installation options. This option includes the type 2 driver and its name is why this driver is sometimes referred to as the "client" driver.

Cluster Notes

If you are installing a cluster, be sure to install the driver in the same location on each machine on which you plan to install the WebLogic Server software—that is, use the same path name and directory name on each machine.

Oracle Installations: Installing the Type 4 (Thin) JDBC Driver

The type 4 JDBC driver is installed during the Oracle installation on the database host machine. It is located in a zip file named `classes12.zip` that also contains all of the Oracle JDBC drivers. Note that the zip file named `nls_charset12.zip` file is located in the same directory. You need this file if you plan to support languages other than English or if you plan to use the CS-Desktop feature.

If you do not intend to install WebLogic on the database host machine, copy the `classes12.zip` file and `nls_charset12.zip` file to any directory on each application server machine.

To "install" the Type 4 driver, complete the following steps:

1. Copy both the `classes12.zip` file and `nls_charset12.zip` file from the database host machine to the host that will run WebLogic. They are located in the following directory:

   oraHome`/jdbc/lib`

   You can copy them into any directory on that host.

2. Do **not** unzip the files.
3. Record the names of these directory in the *JDBCdir* row of Table 11, “JDBC Parameters” on page 26.

Cluster Notes

When you are installing a cluster, be sure to install the driver in the same location on each machine on which you plan to install the WebLogic Server software—that is, use the same path name and directory name on each machine.

Next Step

Proceed to one of the following two chapters to install the appropriate web server:

- Chapter 6, “Installing IIS on Windows”
- Chapter 7, “Installing Apache on Solaris or Linux”

Section 3

Web Server

This section describes how to install a Web server. It contains the following chapters:

- Chapter 6, “Installing IIS on Windows”
- Chapter 7, “Installing Apache on Solaris or Linux”

Chapter 6

Installing IIS on Windows

This chapter describes how to install and test Microsoft's Internet Information Services (IIS). It contains the following sections:

- Install IIS
- Document Your IIS Installation
- Verify the Installation
- Next Step

Typically, IIS is either fully or partially installed on most Windows 2000 machines:

- If IIS is fully installed, start with the section "Document Your IIS Installation" on page 46.
- If IIS is only partially installed, start with the first section, "Install IIS" on page 46.

Install IIS

If IIS is not installed or is only partially installed, follow Microsoft's instruction for installing IIS on a Windows 2000 system.

As a convenience, here is a quick synopsis of the instructions:

1. Select **Start** > **Settings** > **Control Panel**.
2. Select **Add/Remove Programs**.
3. Select the **Add/Remove Windows Components** tab on the left.

   The **Add/Remove Windows Components Wizard** appears.
4. Select **Internet Information Services (IIS)** and then follow the instructions for installing it.

Document Your IIS Installation

We strongly recommend that you document the details of your IIS installation in Table 4, "Web Server Parameters" on page 23. The following information will help you complete this table:

Table 14: IIS Parameters

Parameter	What it Holds
Web Version	The version number of the IIS software that you installed.
Web Host Name (`WebHost`)	The name by which the installation machine is known on the network.
Web Host IP Address (`WebIP`)	The numeric Internet Protocol address assigned to the web server host machine.
Web Server Port (`WebPort`)	The port number assigned for web server communications. By default, it has the value 80.

Verify the Installation

After you have installed IIS, you start it and then browse to it in a web browser to determine whether it is serving pages as it should.

Start IIS

You can start the various IIS services in various ways. To be sure that all the necessary services are running, start IIS from the **Services** node. Complete the following steps:

1. Right-click on the **My Computer** icon.
2. Select **Manage** from the right mouse menu.

3. In the **Computer Management** dialog box, expand the **Services and Applications** node in the tree.
4. Select **Services**.
5. In the list of services on the right, right click **IIS Admin Service**.
6. Select **Start** from the right mouse menu.

To start or stop the default web site only, complete the following steps:

1. Right-click on the **MyComputer** icon.
2. Select **Manage** from the right mouse menu.
3. In the **Computer Management** window, expand the **Services and Applications**. node in the tree.
4. Expand the **Internet Information Services** node.
5. Right-click on **Default Web Site**.
6. Select **Start** or **Stop**, as appropriate, from the right mouse menu.

Verify that IIS is Serving Pages

To verify that IIS can serve pages, test it from both the server that is hosting it and from another browser on the network.

Complete the following steps:

1. Start a browser on the host on which IIS is running.
2. From the browser, go to the following URL:

   `http://WebHost:WebPort`

3. Do one of the following:
   - If the browser displays the IIS home page, then IIS is installed and running properly. Continue to step 4.
   - If the browser returns an error, consult Microsoft's documentation, determine what went wrong, and fix it before you continue.
4. Start a browser on another machine on your network (a host other than the machine hosting IIS).
5. From the browser, go to the following URL:

   `http://WebHost:WebPort`

   If the browser displays the IIS "Under Construction" page, then IIS is installed and running and the network naming service appears to be working properly.

Next Step

Proceed to Chapter 8, "Installing BEA WebLogic Server."

Chapter 7

Installing Apache on Solaris or Linux

This chapter describes how to install and configure Apache HTTP Server. As previously mentioned, you can install Apache on the same machine that will host WebLogic and Content Server or you can install and use it on a separate host.

This chapter contains the following sections:

Install Apache

Apache HTTP Server is often pre-installed on Solaris 8, Solaris 9, Linux RedHat, and Linux SuSE systems. Determine whether Apache is installed on the system(s) on which you plan to run your web server(s).

If it already installed, continue with "Document Your Apache Parameters" on page 50.

If it is not already installed, you can do one of the following:

- Install it from your Solaris or Linux CD.
- Build it from source; that is, select the modules and compile the Apache executable yourself.

If you want to build it from source, follow the instructions in Appendix A, "Building the Apache Web Server" and examine the information that Apache makes available at `http://httpd.apache.org/`

Document Your Apache Parameters

We strongly recommend that you document the details of your Apache installation in Table 4, "Web Server Parameters" on page 23. The following information will help you complete this table:

Table 15: Apache Parameters

Parameter	What it Holds
Web Server Version (*`WebVersion`*)	The version of Apache that the host is running. Note that you must use a version that Content Server supports.
Web Host Name (*`WebHost`*)	The name by which the Apache host machine is known on the network.
Web Host IP Address (*`WebIP`*)	The numeric Internet Protocol address assigned to the Apache host machine.
Web Server Port (*`WebPort`*)	The port number assigned for Apache communications. By default, it has the value 80.
Apache Root Directory (*`ApacheRoot`*)	The top-level directory in which Apache is installed. Immediate subdirectories of *`ApacheRoot`* include `bin` and `conf`.

Verify that Apache Contains the Correct Module

Apache is modular software, built from a set of modules. WebLogic Server requires that the `mod_so.c` module be present on the machine that is hosting the Apache web server. Please verify that your Apache server contains this module by using the command `httpd` with the `-l` option and search for `mod_so` in the output.

For example:

```
$ ApacheRoot/bin/httpd -l | grep 'mod_so'
mod_so.c
```

Examine the output and do one of the following:

- If the output from the preceding command contains `mod_so.c`, then your version of Apache contains the correct module. Proceed to "Verify that Apache Runs Properly" on page 51.
- If the output from the preceding command does not contain `mod_so.c`, you must rebuild and reinstall Apache. See Appendix A, "Building the Apache Web Server" for details and then return to this chapter.

Verify that Apache Runs Properly

Next, start Apache and verify that it is running properly. Complete the following steps:

1. Using a UNIX or Linux shell, change directory to the *`ApacheRoot`* directory. For example:

   ```
   $ cd /local/apache
   ```

2. Inside that directory, you'll find a subdirectory named `bin`. Change to that subdirectory:

   ```
   $ cd bin
   ```

3. Run the following command to start Apache. Note that if you are starting Apache at port 80 (or any port less than 1024), you must be logged in as `root`.

   ```
   $ ./apachectl start
   ./apachectl start: httpd started
   ```

4. In a browser, go to the following URL:

   `http://WebHost:WebPort`

   If the browser displays a "Powered by Apache" page, then your installation was successful.

Next Step

After installing, documenting, and verifying Apache, you must now install WebLogic. Proceed to Chapter 8, "Installing BEA WebLogic Server."

Section 4

Application Server

This section describes how to install and configure WebLogic. It contains the following chapters:

Chapter 8

Installing BEA WebLogic Server

This chapter explains how to install and configure BEA WebLogic Server versions 7.1 and 8.1. It contains the following sections:

About Cluster Installations

Because there are so many ways that you can set up a WebLogic cluster, we cannot provide definitive instructions for cluster installations. For example, you could install the cluster's admin server on a separate host or keep it on the primary cluster member. And that is only one of the many variables.

What we can do in this guide is provide tips and describe one basic methodology. But even that basic methodology is complicated by the fact that there are procedural differences depending on whether you are installing WebLogic 8.1 or WebLogic 7.1.

With WebLogic 7.1, if you create the WebLogic cluster first and then install WebLogic and Content Server on all the cluster members, the name of the cluster appears in your URLs. For this reason, if you are installing a 7.1 cluster, we suggest that you install WebLogic and Content Server on all the cluster members first, then create the cluster, and then configure the cluster.

With WebLogic 8.1, this issue with the cluster name and the URLs has been fixed. If you are installing an 8.1 cluster, you can create the cluster first and then install WebLogic and Content Server on all the cluster members.

Tips for All Cluster Installations

No matter which version of WebLogic you are using, the following conditions and tips are true:

- You complete a full installation on each cluster member before moving to the next. That is, you install WebLogic, Content Server, and the CS content applications on each cluster member before moving to the next.
- You use one connection pool and one datasource for all the machines in the cluster. (But of course you install a JDBC driver on each application server host.)
- The following directories and names should be the same for each cluster member:
  - The location of the JDBC driver
  - The path to and name of the `WL_Home` directory.
  - The WebLogic domain name that represents Content Server
  - The path to the WebLogic domain name

Requirements for All Cluster Installations

Before you begin installing WebLogic on the first server in your cluster, be sure that you have the following information:

- A cluster-enabled license from BEA.
- A multicast address for intercluster communciations.
- The host names and IP addresses of each member of the cluster. Note that they must have fixed IP addresses.
- The server connection port for each member of the cluster.

Basic Methodology for 8.1 Clusters

As mentioned, for 8.1 clusters, you start by installing WebLogic on the primary cluster member and creating the cluster. You create one WebLogic Admin Server to administer all the managed servers in the cluster and you also create managed server instances to represent all the secondary cluster members, indentifying them in the cluster definition.

When you create the connection pool and datasource, rather than identifying the individual managed servers as the target, you target the cluster name itself. And then you install and Content Server and the rest of the Content Server products before you start installing any of the secondary cluster members.

During an installation on a secondary cluster member, you make sure that its managed server name and IP address exactly match the name and IP address that was specified for it in the WebLogic cluster instance.

Basic Methodology for 7.1 Clusters

As mentioned, for 7.1 clusters, we suggest that you install all software on all the cluster members first, and then create the WebLogic cluster instance. If you create the cluster first, the cluster name appears in your URLs.

Primary Cluster Member

On the primary cluster member, you create one Admin server and you create a managed server for each secondary cluster member. When you create the connection pool and datasource, you target the managed server on the primary cluster member only. And then you install and Content Server and the rest of the Content Server products before you start installing any of the secondary cluster members.

Secondary Cluster Members

During an installation on a secondary cluster member, you create its managed server and then identify the admin server on the primary cluster member as its admin server. (Yes, you can do this even when you have not created a cluster.) The managed server instance must have the exact same name, host name, and IP address that you used to identify it on the primary cluster member.

Do not create a datasource and connection pool for the managed server on a secondary cluster member. Instead, return to the primary cluster member and add the secondary cluster member's managed server as a target to both the datasource and connection pool that you created for the admin server.

Post-Installation Configuration Tasks for 7.1 Clusters

Following this methodology means that there are several additional post-installation configuration steps to take after you have completed the installation of WebLogic, Content Server, and the Content Server products on each cluster member. For example:

- Create the WebLogic cluster instance.
- Edit the Content Server web application: set the cluster as the target and remove the managed servers as targets.
- If you are using the Apache HTTP Server, on each web server, edit the `httpd.conf` file: change the `IfModule mod_weblogic.c` statement so that it identifies the cluster rather than an individual WebLogic host. For example:

  ```
  <IfModule mod_weblogic.c>
     WeblogicCluster 10.100.31.93:10001,10.100.31.94:10001
  </IfModule>
  ```

  Note that you include the IP address or machine name and the port number of each managed server in the cluster and separate them with commas as shown in the example above.
- Edit the connection pool: set the cluster as the target and remove the managed servers on the secondary cluster members as targets.
- Edit the datasource: set the cluster as the target and remove the managed servers on the secondary cluster members as targets.
- Note that you do **not** change the startup scripts that identify the local JDBC driver on each cluster member.

- On each cluster member, edit the `weblogic.xml` file to enable session replication through memory (multicast). Add the following statement:

```
<session-descriptor>
   <session-param>
         <param-name>PersistentStoreType</param-name>
         <param-value>replicated</param-value>
   </session-param>
</session-descriptor>
```

Then you restart all the admin and managed servers in the cluster.

Before You Begin

Before you begin installing BEA WebLogic Server, be sure that you complete the tasks in this section.

Read the WebLogic Documentation

Go to the BEA e-docs web site and examine their installation materials:

`http://e-docs.bea.com`

Which User?

On Solaris and Linux, we recommend installing WebLogic while you are logged in as a nonroot user. It is typical to create a new user for this purpose. We refer to the user who installs WebLogic and Content Server as the `csuser`.

Record the following information about the csuser in Table 5, “Who Installed WebLogic?” on page 24:

- In the *`installerName`* row, record the username.
- In the *`installerPass`* row, record the password.

Remember that you must install Content Server while you are logged in as this same user.

Cluster Installations

You must examine BEA’s documentation on cluster installs and obtain the proper licenses. It is also a good idea to draw a map of your cluster, identifying each member, its IP address and hostname, and its managed server or admin server name.

Additionally, be sure that you read the sections “Tips for All Cluster Installations” on page 56 and “Requirements for All Cluster Installations” on page 56 as well.

Synchronize Clocks

You must synchronize the internal system clocks on all the machines that are members of the cluster. If you do not, the system can suffer from problems with synchronizing processes across cluster members.

Note

We recommend that you set up an automated or manual process that periodically synchronizes system clocks daily or weekly, depending on the accuracy of your system clocks.

Installing WebLogic Server

Your first resource for insallation instructions should be BEA's documentation for installing WebLogic 8.1 or 7.1, as appropriate for your system.

Note for Linux Installations

If the WebLogic installer does not function correctly—it does not appear, it becomes unresponsive, or something similar—set the following environmental variable in the WebLogic start script:

```
LD_ASSUME_KERNEL=2.4.3; export LD_ASSUME_KERNEL
```

As a convenience, the following tables provide suggested answers to the installation prompts. These tables describe the simplest possible case: a single, non-clustered installation. These tables do not describe how to configure WebLogic clusters. You must consult the WebLogic documentation and follow it when you are installing a WebLogic cluster.

There are separate tables for WebLogic 8.1 and 7.1. Consult the version that is appropriate for your system.

WebLogic 8.1 Installations

Table 16: Suggested Replies to WebLogic 8.1 Installation Screens

Installation Screen	Selection
Welcome	**Next**
License Agreement	**Yes**
Home Directory	Accept the default and note it in the *beaRoot* row of Table 6, "WebLogic Installation Parameters" on page 24.
Choose Install Type	**Custom Installation**
Choose Components	Select **WebLogic Server** and make sure that its subentries (**Server**, **Workshop**, and **Server Examples**) are also selected. Optionally, select anything else you want.
Choose Product Installation Directory	Accept the default and record it in the *WL_Home* row of Table 6, "WebLogic Installation Parameters" on page 24.

Installation Screen	Selection
Install Node Manager	**No**
Installer	The installer will run, which will take several minutes. At the end of the installation, select **Next**.
Installation Complete	Click **Done**. The **Configuration Wizard** appears.
Configuration Wizard, Create a Configuration	**Create a new WebLogic configuration**
Select a Configuration Template	**Basic WebLogic Server Domain** Then, enter a domain name, for example, `csdomain`. Record the domain name in the *`csDomain`* row of Table 7, "WebLogic Admin Server Parameters" on page 24.
Express or Custom Configuration	**Custom**
Administration Server Configuration (Enter your selections in Table 7, "WebLogic Admin Server Parameters" on page 24.)	**Server Name**: `csadmin`
	Listen Address: `localhost`
	Listen Port: `7001`
	SSL Port: `7002`
Multiple Servers, Clusters, and Managed Options	**Yes**
Managed Server Configuration (Enter your selections in Table 8, "WebLogic Managed Server Parameters" on page 25.)	**Server Name**: `csmanaged`
	Listen Address: *`name_of_your_host`* or `localhost`
	Listen Port: `8001`
	SSL Port: `8002`
Cluster Configuration	**Next** However, if you are installing a cluster, consult the WebLogic installation documentation for assistance.
Machine Configuration	**Next**
Database (JDBC) Options	**Skip**
Messaging (JMS) Options	**Skip**
Advance Security Options	**Skip**
Configure Username and Password (Enter your selections in Table 7, "WebLogic Admin Server Parameters" on page 24.)	**Username:** *`AdminUserName`*
	Password: *`AdminUserPassword`*
Windows Options (Windows Only)	**Create Start Menu:** yes **Install Administrative Server as a service:** no

Installation Screen	Selection
Build Start Menu Entries (Windows Only)	Accept the defaults.
WebLogic Configuration Environment	**WebLogic Configuration Startup Mode:** Production Mode **Java SDK Selection:** BEA Supplied SDKs, Sun SDK 1.4.1_02-ea
Create WebLogic Configuration	Enter a name in the **Configuration Name** field, which is located in lower right corner. Record this name in the *csDomain* row of Table 7, "WebLogic Admin Server Parameters" on page 24. Click **Create**
Installation Complete	**Done**

Continue to "Verify the Installation" on page 62.

WebLogic 7.1 Installations

Table 17: Suggested Replies to WebLogic 7 Installation Screens

Installation Screen	Selection
Welcome	**Next**
License Agreement	**Yes**
Home Directory	Accept the default and note it in the *beaRoot* row of Table 6, "WebLogic Installation Parameters" on page 24.
Choose Install Type	**Custom Installation**
Choose Components	Select **WebLogic Server** and make sure that its subentries (**Server**, **Workshop**, and **Server Examples**) are also selected. Optionally, select anything else you want.
Choose Product Installation Directory	Accept the default and record it in the *WL_Home* row of Table 6, "WebLogic Installation Parameters" on page 24.
Installer	The installer will run, which will take several minutes. At the end of the installation, just select **Next**.
Run Configuration Wizard	**Yes**
Choose Domain Type and Name	Select `WLS Domain`. Then, enter a domain name, for example, `csdomain`. Record the domain name in the *csDomain* row of Table 7, "WebLogic Admin Server Parameters" on page 24.
Choose Server Type	**Admin Server with Managed Server(s)**

Installation Screen	Selection
Choose Domain Location	Accept the default
Configure Standalone/Administrative Server (Enter your selections in Table 7, "WebLogic Admin Server Parameters" on page 24.)	**Server Name**: `csadmin`
	Listen Address: `localhost`
	Listen Port: `7001`
	SSL Port: `7002`
Configure Managed Servers in Admin Server (Enter your selections in Table 8, "WebLogic Managed Server Parameters" on page 25.)	**Server Name**: `csmanaged`
	Listen Address: *`name_of_your_host`* or `localhost`
	Listen Port: `8001`
	SSL Port: `8002`
Create Administrative User (Enter your selections in Table 7, "WebLogic Admin Server Parameters" on page 24.)	**Username:** *`AdminUserName`*
	Password: *`AdminUserPassword`*
Configuration Wizard Complete	**End Configuration Wizard.**
Installation Complete	**Done**

Verify the Installation

After installing, you should verify the installation by performing the following steps:

- Set the Login Name and Password in the Startup Script (Optional)
- Start the WebLogic Admin Server
- Verify the Installation

Set File Encoding for UTF-8

Just as the database character set must support all the characters that you intend to store, the application server file encoding setting should also be set appropriately. If you plan to use the CS-Desktop feature, you must set the file encoding property in the `startManagedWebLogic.sh` or `startManagedWebLogic.cmd` script to UTF-8.

Complete the following steps:

1. Navigate to the directory that contains the `startManagedWebLogic.cmd` or `startManagedWebLogic.sh` file. Typically it is:

   `beaRoot``\user_projects\`*`csdomain`*

   You recorded the value for *`beaRoot`* and *`csDdomain`* (which may be `csdomain`) in Table 6, "WebLogic Installation Parameters" on page 24.

2. Open the file in a text editor such as vi or Notepad.

3. Scroll down to the `JAVA_OPTIONS` section. At the beginning of the statement, insert:

   `-Dfile.encoding=UTF-8`

For example:

```
JAVA_OPTIONS="-Dfile.encoding=UTF-8
Dweblogic.security.SSL.trustedCAKeyStore=/local01/bea702/
weblogic700/server/lib/cacerts"
```

Note that there is a space that separates the `UTF-8` setting from the rest of the string.

4. Save and close the file.

Set the Login Name and Password in the Startup Script (Optional)

While experimenting with the WebLogic admin server, you might need to restart it many times. By default, WebLogic will prompt you for a login name and password every time you restart. If this becomes annoying, you can take the following steps to embed the login and password information in the startup script (a configuration file). After you embed it in this script, WebLogic will not prompt you for a login and password, thus saving some time and aggravation.

Caution

Embedding a password in a plain text file is a gaping security hole. Never do this for a live site.

To provide the login name and password in the startup script, complete the following steps:

1. Navigate to the directory that contains the `startWebLogic.cmd` or `startWebLogic.sh` file. Typically it is:

   beaRoot`\user_projects\`*csdomain*

   You recorded the value for *beaRoot* and *csDdomain* (which may be `csdomain`) in Table 6, “WebLogic Installation Parameters” on page 24.

2. Open the file in a text editor such as vi or Notepad.

3. Insert the following values:

   ```
   set WLS_USER = wlAdminName
   set WLS_PW = wlAdminPass
   ```

   You recorded the values of *wlAdminName* and *wlAdminPass* in Table 7, “WebLogic Admin Server Parameters” on page 24.

4. Save and close the file.

5. Repeat these steps for the `startManagedWebLogic.sh` or `startManagedWebLogic.cmd` file.

Start the WebLogic Admin Server

To start the WebLogic Admin Server on a Windows system, you can either click the **startWebLogic** icon or invoke the `startWebLogic.cmd` file from a DOS prompt.

To start the WebLogic Admin Server on a Solaris or Linux system, invoke the `startWebLogic.sh` script.

The `startWebLogic.sh` or `startWebLogic.cmd` scripts are located at:

beaRoot\user_projects*csdomain*

You recorded the value for *beaRoot* and *csDdomain* (which may be `csdomain`) in Table 6, "WebLogic Installation Parameters" on page 24.

After it is invoked, WebLogic displays a variety of messages in a console window. WebLogic is sucessfully installed and running when this final message appears:

```
<Server started in RUNNING Mode>
```

Verify the Server Installation

After starting WebLogic, open the Admin Console and verify that the admin and managed servers that you created during the installation were created correctly.

Complete the following steps:

1. Open a browser and enter the following URL:

   http://*WebHost*:*wlAdminPort*/console

   For example:

   http://MyWebHost:7001/console

   The admin server login page appears.
2. In the admin server login page, log in as the WebLogic administrator user that you created when you installed WebLogic and then recorded in Table 7, "WebLogic Admin Server Parameters" on page 24.
3. In the Admin Console, browse down the tree to the domain that you created for Content Server during the WebLogic installation (*csDomain*).
4. Select ***csDomain*** > **Servers** > ***AdminServerName***.
5. Verify that the servers that you specifed during the WebLogic installation are listed.
6. Select the name of the managed server.
7. In the **Configuration** form, select the **Deployment** tab.
8. In the **Staging Mode** field, select **nostage** from the drop-down list.
9. Click **Apply**.

Next Step

After installing WebLogic, proceed to one of the following chapters:

- Chapter 9, "Configuring IIS for WebLogic and Content Server"
- Chapter 10, "Configuring Apache for WebLogic and Content Server".

Chapter 9

Configuring IIS for WebLogic and Content Server

After you have installed both IIS and WebLogic, you configure IIS to interact with WebLogic and Content Server.

- You configure IIS for WebLogic by setting up the ISAPI plugin that WebLogic provides.
- You configure IIS for Content Server by creating a web root and document directory and then identifying the location of the document root with an IIS virtual directory..

This chapter contains the following sections:

If you are installing on Solaris, skip this chapter and proceed to Chapter 10, “Configuring Apache for WebLogic and Content Server.”

Configure IIS for WebLogic

You configure IIS for WebLogic by mapping two file extensions to the WebLogic application by pointing to a WebLogic-provided `.dll` and by creating an ISAPI filter that uses the WebLogic plugin or filter (`.dll`) for IIS.

Mapping the file extensions takes two steps:

- Use the IIS console to identify the file extensions and the appropriate `.dll` to use for them.
- Create a configuration file called `iisproxy.ini`, which specifies how to contact WebLogic. Creating the configuration file is a manual step that you complete outside of the IIS console.

Create the Application Mappings and the ISAPI Filter

Complete the following steps:

1. Right click on the **My Computer** icon on your desktop and select **Manage** from the right mouse menu.
2. In the Computer Management window, select **Services and Applications** > **Internet Information Services**.
3. Right click on **Default Web Site** and select **Properties**.
4. In the **Default Web Site Properties** dialog box, select the **Home Directory** tab.
5. Click in the **Execute Permissions** field and select **Scripts and Executables** from the drop-down list.
6. Click **Configuration**.
7. In the **Application Configuration** dialog box, select the **App Mappings** tab.
8. In the **App Mappings** dialog box, verify that the **Cache ISAPI applications** option is selected.
9. Click **Add**.
10. In the **Add/Edit Application Extension Mapping** form, create a mapping for the `.jsp` file extension. Enter the following values:

Field	Set It to This Value
Executable	Click **Browse.** Navigate to and select: *`WL_Home`*`\server\bin\iisproxy.dll` You recorded the value for *`WL_Home`* in Table 6, “WebLogic Installation Parameters” on page 24.
Extension	`jsp` (not `.jsp` — do not include the period)
Verbs	**All verbs** (the default)
Script engine	Clear this option.
Check that file exists	Clear this option.

11. Click **OK**.
12. Back in the **App Mappings** dialog box, click **Add** again.
13. This time in the **Add/Edit Application Extension Mapping** dialog box, create a mapping for the `.wlforward` file extension. Enter the following values:

Item	Set It to This Value
Executable	Click **Browse.** Navigate to and select: *`WL_Home`*`\server\bin\iisproxy.dll` **Note**: Be sure to select `iisproxy.dll`; do **not** select `iisforward.dll` You recorded the value for *`WL_Home`* in Table 6, "WebLogic Installation Parameters" on page 24.
Extension	`wlforward` (not `.wlforward` — do not include the period)
Verbs	**All verbs** (the default)
Script engine	Clear this option.
Check that file exists	Clear this option.

14. Click **OK**.
15. Back in the **App Mappings** dialog box, click **Apply**; then click **OK**.

    In the the **Application Configuration** window you see two new **Application Mapping** entries named `.jsp` and `.wlforward`.
16. Click **OK**.
17. In the **Default Web Site Properties** dialog box, select the **ISAPI Filters** tab.
18. Click **Add...**.
19. In the **Filter Properties** form, create a filter that uses the WebLogic `iisforward.dll` file. Enter the following values:

Item	Set It to This Value
Filter Name	You can specify any arbitrary name, but we recommend: `iisforwardfilter` Record your selection in the *`FilterName`* row of Table 4, "Web Server Parameters" on page 23.
Executable	Click **Browse.** Navigate to and select: *`WL_Home`*`\server\bin\iisforward.dll` You recorded the value for *`WL_Home`* in Table 6, "WebLogic Installation Parameters" on page 24.

20. Click **OK**.
21. In the **Default Web Site Properties** dialog box, click **Apply;** then click **OK**.

Create the iisproxy.ini Configuration File

Next, create the `iisproxy.ini` file. Complete the following steps:

1. Open a text editor, such as Notepad, and create a new file.
2. In this file, enter the following statements. Be aware that the case of each property must exactly match the case specified here:

   ```
   WebLogicHost=wlHost
   WebLogicPort=wlManagedPort
   ConnectTimeoutSecs=20
   ConnectRetrySecs=5
   WlForwardPath=/servlet
   ```

   You recorded the value of *`wlHost`* Table 6, "WebLogic Installation Parameters" on page 24. (If WebLogic is on the same host system as IIS, you can set *`wlHost`* to `localhost`.) You recorded the value of *`wlManagedPort`* in Table 8, "WebLogic Managed Server Parameters" on page 25.
3. Save and name the file: `iisproxy.ini`
4. Place the file in the following directory:

   `WL_Home``\server\bin`
5. Restart all the IIS services. (See "Start IIS" on page 46 for details on restarting IIS.)

Configure IIS for Content Server

You configure IIS for Content Server by creating a Content Server documentation root and then identifying it to IIS with a virtual directory.

The Content Server installation program creates the Content Server documentation root. However, at this stage in your installation, the directory does not yet exist because you have not yet run the Content Server installation program. In this step, you manually create this directory and then create a virtual directory for it.

Note the following:

- If you installed WebLogic and IIS on the same server, the top level of the Content Server documentation root is also the Content Server installation directory. Be sure that you specify this directory during the Content Server installation.
- If you installed IIS and WebLogic on separate machines, there is an additional step to take after the Content Server installation. After you install Content Server on the application server host, you must copy the files from the Content Server documentation root on that machine to the documentation root on the web server machine.

There are two steps in this task:

- "Create the Content Server Document Root" on page 69
- "Create the IIS Virtual Directories for Content Server" on page 69

Create the Content Server Document Root

You create the Content Server document root on the machine(s) where you installed IIS.

Complete the following steps:

1. Create the top-level directory for the Content Server documentation root. Note that if WebLogic and IIS are installed on the same host, this top-level directory will also be the Content Server installation directory. For example:

   `C:\ContentServer`

   If WebLogic and IIS are installed on the same host, record the name of this folder in the *`csRoot`* row of Table 4, "Web Server Parameters."

2. In this top-level Content Server directory, create a subdirectory called `futuretense_cs`.

   `C:\ContentServer\futuretense_cs`

   Note

   You may place the *`csRoot`* anywhere and give it any name; however, you must name its subdirectory `futuretense_cs`.

3. Then, in the `futuretense_cs` directory, create a subdirectory named `Xcelerate`. It must be spelled exactly, with an initial capital letter "X." For example:

   `C:\ContentServer\futuretense_cs\Xcelerate`

Create the IIS Virtual Directories for Content Server

Follow these steps to configure IIS to use the Content Server document root directory:

1. Select **Start** > **Programs** > **Administrative Tools** > **Internet Services Manager**
2. Expand the node that represents your Content Server system.
3. Click to select the **Default Web Site**.
4. Select **Action** > **New** > **Virtual Directory**.
5. Create a virtual directory for the `futuretense_cs` directory as follows:
   - Name it `futuretense_cs`
   - In the **Web Site Content Directory** window, browse to the `futuretense_cs` directory that you created on this machine.
   - In the **Access Permissions** window, select **Read**, **Execute**, and **Browse**. Do **not** select Run Scripts or Write.
6. Select **Action** > **New** > **Virtual Directory** again and this time create a virtual directory for the `Xcelerate` directory. Name it `Xcelerate`, browse to the `Xcelerate` directory that you created, and give it the same access permissions as for the `futuretense_cs` directory.

Verify the ISAPI Plugin Configuration

To verify that the ISAPI plugin is configured correctly, you must determine whether it is running and is forwarding requests to WebLogic.

Complete the following steps:

1. Open a browser and navigate to the following URL:

   `http://WebHost:WebPort/servlet`

   You recorded the values of *`WebHost`* and *`WebPort`* in Table 4, "Web Server Parameters" on page 23.

If the IIS/WebLogic configuration is working properly, you should see an **error page returned by the ISAPI plugin.** If you do not see the ISAPI error page, the plugin is not configured correctly. Examine all of your application mappings and the `iisproxy.ini` file. Typically, the case of a property or an incorrect value in the `iisproxy.ini` is the cause.

Next Step

Proceed to Chapter 11, "Configuring the JDBC Driver."

Chapter 10

Configuring Apache for WebLogic and Content Server

You configure Apache for WebLogic by installing the Apache plugin that WebLogic provides, and then editing the Apache configuration file so that the plugin module is loaded when Apache is started.

You configure Apache for Content Server by creating the Content Server document root and then identifying the location of that document root in the Apache configuration file. This step is how you enable Apache to find the Content Server files it needs after you have installed Content Server.

This chapter contains the following sections:

Install the Apache HTTP Server Plugin from WebLogic

The Apache plugin is a shared library that BEA provides so that WebLogic will support Apache. To install it, you obtain it from a WebLogic directory and copy it to the appropriate Apache directory.

If you installed Apache and WebLogic on different hosts, you copy the file from the WebLogic directory on the WebLogic host to the appropriate Apache directory on the Apache host. If you installed Apache on more than one host, you copy the file to the appropriate directory on each host.

Complete the following steps:

1. Do one of the following:
   - If you are using Apache version 2.x, locate the `mod_wl_20.so` file:
   - If you are using Apache version 1.3.x, locate the `mod_wl.so` file:

   For Solaris, the files are located in the *`wl_HOME`*`/server/lib/solaris` directory.

   For Linux, the files are located in the *`wl_HOME`*`/server/lib/linux/i686` directory.

   See Table 6, “WebLogic Installation Parameters” on page 24 for the value of *`wl_HOME`*.
2. Copy the appropriate file to the appropriate directory. Do one of the following:
   - If you are using Apache version 2.x, copy the `mod_wl_20.so` file to: *`ApacheRoot`*`/modules`
   - If you are using Apache version 1.3.x, copy the `mod_wl.so` file to: *`ApacheRoot`*`/libexec`

   You recorded the value of *`ApacheRoot`* in Table 4, “Web Server Parameters” on page 23.

Note that the plugin isn’t completely “installed” until you enter the appropriate `AddModule` and `LoadModule` statements in the `httpd.conf` file, which you will do after you create the Content Server document root.

Create the Content Server Document Root

You configure Apache for Content Server by creating a Content Server documentation root and then identifying it to Apache through alias statements in the `httpd.conf` file.

The Content Server installation program creates the Content Server documentation root. However, at this stage in your installation, the directory does not yet exist because you have not yet run the Content Server installation program. In this step, you manually create this directory and then specify its location in the `httpd.conf` file.

Note the following:

- If you installed WebLogic and Apache on the same server, the top level of the Content Server documentation root is also the Content Server installation directory. Be sure that you specify this directory during the Content Server installation.

- If you installed Apache and WebLogic on separate machines, there is an additional step to take after the Content Server installation. After you install Content Server on the application server host, you must copy the files from the Content Server documentation root on that machine to the documentation root on the web server machine.

You create the Content Server documentation root on the machine(s) that host Apache. Complete the following steps:

1. Create the top-level Content Server directory. If you installed Apache and WebLogic on the same server, this directory is also the Content Server installation directory. For example:

   ```
   $ mkdir /local/ContentServer
   ```

   If you installed Apache and WebLogic on the same host,record the directory in the *csRoot* row of Table 12, “Content Server Configuration” on page 27.

2. Under the *csRoot* directory, create a subdirectory named `futuretense_cs`. For example:

   ```
   $ mkdir /local/ContentServer/futuretense_cs
   ```

   Note

   Although you can name the *csRoot* anything you want, you must name its subdirectory `futuretense_cs`.

   Record the directory in the *csDocRoot* row of Table 12, “Content Server Configuration” on page 27.

3. Under the `futuretense_cs` subdirectory, create a subdirectory named `Xcelerate`. It must be spelled exactly, with an initial capital letter “X.” For example:

   ```
   $ mkdir /local/ContentServer/futuretense_cs/Xcelerate
   ```

Edit the Apache Configuration File

After you have installed WebLogic’s Apache plugin and have created the Content Server document root, edit the Apache configuration file so that Apache has the information it needs to load the plugin and to find the Content Server document root. You must edit the `httpd.conf` file for each web server in your system.

Complete the following steps:

1. As the **root user**, open the `httpd.conf` file in a text editor such as vi or emacs. It is located at one of the following one of the following pathnames:

   ApacheRoot`/conf/httpd.conf`

   or

   `/etc/Apache/httpd.conf`

   You recorded the value of *ApacheRoot* in Table 4, “Web Server Parameters” on page 23.

2. Scroll down to the `Dynamic Shared Object` section. Under the LoadModule example, do one of the following:
   - If you are using Apache version 1.3.x, add the following directive:
     `LoadModule weblogic_module libexec/mod_wl.so`
   - If you are using Apache version 2.x, add the following directive:
     `LoadModule weblogic_module modules/mod_wl_20.so`
3. If you are using Apache version 1.3.x, scroll down to the last `AddModule` directive, and add the following directive:

   ```
   AddModule mod_weblogic.c
   ```

4. For both Apache version 1.3.x and 2.x, scroll down to the last `<IfModule>` directive, and add the following directive under it:

   ```
   <IfModule mod_weblogic.c>
     WebLogicHost wlHost
     WebLogicPort wlManagedPort
     HungServerRecoverSecs 7200
   </IfModule>
   ```

   You recorded the value of *`wlHost`* in Table 6, “WebLogic Installation Parameters” on page 24 and the value of *`wlManagedPort`* in Table 8, “WebLogic Managed Server Parameters” on page 25.
5. For both Apache version 1.3.x and 2.x, scroll down to the last `<Location>` directive and add the following directive:

   ```
   <Location /servlet>
     SetHandler weblogic-handler
   </Location>
   ```

6. Search for the `Alias` section in this file. After the `Alias /icons/` statement, identify the `futuretense_cs` and the `Xcelerate` directories with two new Alias statements. Use the absolute pathname of the Content Server document root directory `futuretense_cs` and to the `Xcelerate` subdirectory.

   For example, if your Contente Server document root directory is `/local/ContentServer/futuretense_cs`, you would insert the following directives into the `Alias` section:

   ```
   Alias /futuretense_cs "/local/ContentServer/futuretense_cs"
   <Directory "/local/ContentServer/futuretense_cs"
   Options Indexes MultiViews
   AllowOverride None
   Order allow,deny
   Allow from all
   </Directory>

   Alias /Xcelerate "/local/ContentServer/Xcelerate"
   <Directory "/local/ContentServer/futuretense_cs/Xcelerate"
   Options Indexes MultiViews
   AllowOverride None
   Order allow,deny
   Allow from all
   </Directory>
   ```

7. Save and close the `httpd.conf` file.

Restart and Verify Apache

After you finish modifying the `httpd.conf` file, restart Apache so the configuration changes are implemented and then verify that it is running properly. Complete the following steps:

1. In a UNIX or Linux shell, change directories as follows:

   $ **cd** *ApacheRoot*/**bin**

   The Apache start and restart scripts are at *ApacheRoot*/bin/apachect1

2. Enter the following command:

   $ **apachectl restart**

3. In your browser, go to the following URL:

   http://*WebHost*:*WebPort*

   You recorded the values of *WebHost* and *WebPort* in Table 4, "Web Server Parameters" on page 23.

4. If your browser displays the following question, then you have properly configured Apache to work with WebLogic:

   "Seeing this instead of the website you expected?"

   If you do not see the preceding message, refer to BEA's documentation for help.

Start the WebLogic Managed Server and Verify the Plugin Configuration

You started the WebLogic Admin server at the end of Chapter 8, "Installing BEA WebLogic Server." Now start the WebLogic Managed Server and check that WebLogic's Apache plugin is routing requests to the WebLogic managed server (which will become the Content Server managed server after you install Content Server).

Complete the following steps:

1. Open the `startManagedWebLogic.sh` script using vi or another text editor. It is located here:

   $ *beaRoot*/user_projects/*csDomain*/startManagedWebLogic.sh

   You recorded the value for *beaRoot* and *csDomain* (which may be csdomain) in Table 6, "WebLogic Installation Parameters" on page 24.

2. Scroll down the file, and just after the set `WL_HOME` statement, insert the following lines:

   set ADMIN_URL=http://*wlhost*:*wlAdminPort*
   set SERVER_NAME=*wlManagedSerName*

   You recorded the values for *wlhost*, *wlManagedSerName* and *wlAdminPort* in the tables in the section named "WebLogic Parameters" on page 24.

3. Save the file.
4. Invoke the `startManagedWebLogic.sh` script.

   The WebLogic Managed Server displays a variety of messages in a console window. The Managed Server is sucessfully running when this final message appears:

   ```
   <Server started in RUNNING mode.>
   ```

5. Using a browser, go to the following URL:

   `http://`*`WebHost`*`:`*`WebPort`*`/servlet`

   You recorded the values of *`WebHost`* and *`WebPort`* in Table 4, "Web Server Parameters" on page 23.

   If the configuration of WebLogic's Apache plugin is correct, you should see a **404 Page Not Found** error page **returned by WebLogic.** If you see another error message —"The Page cannot be displayed," "Cannot connect to server," an Apache timeout message, and so on—the plugin was not configured correctly.

Next Step

Proceed to Chapter 11, "Configuring the JDBC Driver."

Chapter 11

Configuring the JDBC Driver

This chapter explains how to configure the JDBC driver that you installed after you installed your database. You configure the JDBC driver by creating a JDBC connection pool for WebLogic, creating a data source for Content Server, identifying the `csdbuser` to WebLogic, and then specifying the location of the driver in the appropriate WebLogic startup script.

This chapter contains the following sections:

Remember that there must be a driver installed on each server that hosts the WebLogic server. If you did not install WebLogic on the same server that hosts the DBMS and you have not yet installed the appropriate JDBC driver on the WebLogic host machine, return to "Install the JDBC Driver on the Application Server Hosts" on page 39 and follow the instructions that are appropriate for your system.

Create the JDBC Connection Pool

Cluster Notes

You create one JDBC connection pool per cluster.

For WebLogic 8.1 clusters, you set the cluster as the target of the connection pool.

For WebLogic 7.1 clusters, when you plan to create the WebLogic cluster instance after installing all the software, these are the basic steps for configuring the connection pool:

- You create the connection pool during your installation on the primary cluster member.
- During the installation on each secondary cluster member, you add that managed server to the pool as a target.
- After you create the WebLogic cluster, you edit the targets for the connection pool. You remove the individual managed servers as targets and replace them with the cluster.

WebLogic uses the connection pool to communicate with the DBMS. Complete the following steps to create one:

1. Open the WebLogic Console by entering the following URL in your browser:

   `http://wlHost:wlAdminPort/console`

   You recorded *`wlHost`* in Table 6, “WebLogic Installation Parameters” on page 24 and *`wlAdminPort`* in Table 8, “WebLogic Managed Server Parameters” on page 25.
2. In the WebLogic Console, click the plus sign (+) next to the Content Server domain. You recorded the domain name in the *`csdomain`* row of Table 6, “WebLogic Installation Parameters” on page 24.
3. Click the plus sign (+) next to **Services**.
4. Click the plus sign (+) next to **JDBC**.
5. Click **Connection Pools**.
6. Click the **Configure a new JDBC Connection Pool** hyperlink. A form appears.
7. Enter values for the fields in the form as follows:

Field	Value
Name	Pick any name (such as, `CSpool`) and record this name in the *`poolName`* row of Table 11, “JDBC Parameters” on page 26. This is an arbitrary name that identifies the database connection pool.

Field	Value
URL	Enter the JDBC Connection String (*`connString`*) value. For **JSQL Connect** (SQL Server), your connection string has the following format: `jdbc:JSQLConnect://`*`dbHost`*`:`*`dbPort`* You recorded the values for *`dbHost`* and *`dbPort`* in Table 1, "DBMS Installation Parameters" on page 22 For **Oracle type 2**, your connection string has the following format: `jdbc:oracle:oci8:@`*`Net8ConnectionName`* For example, if your `Net8ConnectionName` is `oraserv1`, your connnection string is `jdbc:oracle:oci8:@oraserv1`. For **Oracle type 4** (thin), your connection string has the following format: `jdbc:oracle:thin:@`*`dbHost`*`:`*`dbPort`*`:`*`sid`* See Table 1, "DBMS Installation Parameters" on page 22, for the values of *`dbHost`* (Database Host Name), *`dbPort`* (Database Port Address), and `sid` (Database Identifier). For example, if you used the sample values, your connection string would be the following: `jdbc:oracle:thin:@centralserve:1521:csdb` Record the value of your JDBC connection string in Table 11, "JDBC Parameters" on page 26.

Driver Classname	For all Oracle drivers, enter the following:
	`oracle.jdbc.driver.OracleDriver`
	For the SQL Server 2000 driver enter the following:
	`com.jnetdirect.jsql.JSQLDriver`

Field	Value

Properties	Set values for the properties that are required for your JDBC driver and WebLogic version:
	• For WebLogic 7.1, regardless of the driver type, you set the `user` and `password` properties to identify the login name and password for the `csdbuser` account.
	• For WebLogic 8.1, you are not required to set the password of the `csdbuser` account with a property—you can use the **Password** field. (See below.)
	• If you are using one of the Oracle drivers, you must specify values for the `dll` and `protocol` properties.
	• If you are using the JSQL Connect driver, you must specify a value for the `database` property.
	JSQL Connect example:
	`user=`*`csdbuser`*
	`password=`*`csdbpass`*
	`database=`*`databaseName`*
	You recorded *`connString`* in Table 11, "JDBC Parameters" on page 26.
	Oracle Type 2 (client) example for Oracle 8.x:
	`user=`*`csdbuser`*
	`password=`*`csdbpass`*
	`dll=ocijdbc8`
	`protocol=oci8`
	Oracle Type 2 (client) example for Oracle 9.x:
	`user=`*`csdbuser`*
	`password=`*`csdbpass`*
	`dll=ocijdbc9`
	`protocol=oci8`
	(Note: yes, the `protocol` is `oci8` for Oracle 9.x)
	Oracle Type 4 (thin) example for Oracle 8.x:
	`user=`*`csdbuser`*
	`password=`*`csdbpass`*
	`dll=ocijdbc8`
	`protocol=thin`
	Oracle Type 4 (thin) example for Oracle 9.x:
	`user=`*`csdbuser`*
	`password=`*`csdbpass`*
	`dll=ocijdbc9`
	`protocol=thin`

ACLName	Leave this field as is or consult BEA's documentation for more information. The value you set for this field will not affect Content Server.
Password	For 7.1, leave this field as is. For 8.1, specify the password of the `csdbuser` account. Then be sure that you are not specifying the `password` property in the property section.

8. Click **Create** to save your changes.
9. Click the **Connections** tab. A form appears.
10. Enter tuning values in the fields as follows. The values listed here are recommended for a starting configuration:

Field	Suggested Value
Initial Capacity	`10`
Maximum Capacity	Set this value to at least `100`.
Allow Shrinking (Checkbox)	Select this option.
All other fields	Leave these fields as is or consult BEA's documentation for more information. The value you set for these fields does not affect Content Server.

11. Click **Apply** to save your changes.
12. Click the **Targets** tab.
13. In the **Available** list, click on the name of the server that represents Content Server (the managed server), which is recorded in the *`wlCSName`* row of Table 9, "WebLogic Content Server Parameters" on page 25.
14. Click the right arrow to move the server from **Available** to **Chosen**.
15. Click **Apply** to save your changes.

Create the Data Source

Cluster Notes

You create one datasource per cluster.

For WebLogic 8.1 clusters, you set the cluster as the target of the data source.

For WebLogic 7.1 clusters, when you plan to create the WebLogic cluster instance after installing all the software, these are the basic steps for configuring the data source:

- You create the data source during the installation on the primary cluster member.
- During the installation on each secondary cluster member, you add that managed server to the datasource as a target.
- After you create the WebLogic cluster, you edit the targets for the datasource. You remove the individual managed servers as targets and replace them with the cluster.

Content Server uses the data source to communicate with the DBMS. Complete the following steps to create one:

1. In the WebLogic Console, click the plus sign (+) next to the domain (*csDomain*).
2. Click the plus sign (+) next to **Services**.
3. Click the plus sign (+) next to **JDBC**.
4. Click **Data Sources**.
5. Click the **Configure a new JDBC Data Source...** hyperlink. The **Create a new JDBC Data Source...** form appears.
6. Enter values for the fields as follows:

Field	Value
Name	Specify an arbitrary name (for example, `fwcs`) and record it in Table 9, "WebLogic Content Server Parameters" on page 25.
JNDI Name	Specify an arbitrary name (for example, `JNDICS`) and record it in Table 9, "WebLogic Content Server Parameters" on page 25. You must provide this name for the "JNDI Data Source Name" during the Content Server installation.
Pool Name	Enter the JDBC Connection Pool Name (for example, `CSpool`) that you created in the procedure "Create the JDBC Connection Pool" on page 78 and then recorded in the *poolName* row of Table 11, "JDBC Parameters" on page 26. You will need to know the pool name that you specified here during the Content Server installation.
All other fields	Leave these fields as is or consult BEA's documentation for more information. The value you set for these fields will not affect Content Server operations.

7. Click **Create**.
8. Click the **Targets** tab in the right pane.
9. In the **Available** list, click the name of the server (*wlCSName*).
10. Click the right arrow to move that server from **Available** to **Chosen**.
11. Click **Apply** to save your changes.

Identify the csdbuser to WebLogic

Cluster Notes

You complete this step once per cluster. During the installation of the primary cluster member, you identify the csdbuser. If you are installing on a secondary cluster member, skip this step.

Now you add the Content Server database user as a WebLogic user. If your DBMS is SQL Server, you created this user with the procedure "Create the Content Server Database User Account (SQL Server)" on page 36. If your DBMS is Oracle, you created this user with the procedure "Create the Content Server Database User Account (Oracle)" on page 37.

Complete the following steps:

1. In the WebLogic Console, click the plus sign (+) next to **Security**.
2. Click the plus sign (+) next to **Realms**.
3. Click the plus sign (+) next to **myrealm**.
4. Click **Users**.
5. Click the **Configure a new User** hyperlink. The **Create a new User** form appears.
6. Enter values for the fields as follows:

Field	Value
Name	Specify the name of the database user that will access Content Server. You recorded this parameter in the `csdbusername` row of Table 3, "Tablespace Parameters" on page 23.
Password	Specify the password of the database user that will access Content Server. You recorded this parameter in the `csdbuserpass` row of Table 3, "Tablespace Parameters" on page 23.

7. Click **Apply** to save your changes.

Edit the Managed Server Startup Script

Finally, you identify the location of the JDBC driver to WebLogic through the startup script for the managed server. Use one of the following procedures, according to your configuration.

SQL Server Installations: Identify the JSQL Connect Driver

Complete the following steps to specify the location of the JSQL Connect driver:

1. Use a text editor like Notepad to open the `startManagedWebLogic.cmd` file.
2. Add the following statement:

   ```
   PRE_CLASSPATH=complete-path-to-the-JSQLConnect-directory/
   JDBC_2.0_Driver/JSQLConnect.jar:$PRE_CLASSPATH
   ```

3. Then, scroll down the file, and just after the set `WL_HOME` statement, insert the following lines:

   ```
   set ADMIN_URL=http://wlhost:wlAdminPort
   set SERVER_NAME=wlManagedSerName
   ```

   You recorded the values for *`wlhost`*, *`wlManagedSerName`* and *`wlAdminPort`* in the tables in the section named "WebLogic Parameters" on page 24.
4. Save and close this file.

Oracle Installations: Identify the Type 2 JDBC Driver

Complete the following steps to specify the location of the type 2 driver:

1. Use a text editor to open the `startManagedWebLogic.sh` (Linux or Solaris) or `startManagedWebLogic.cmd` (Windows) file, as appropriate.
2. After the initial comments and the server name statement, insert the following lines:

   ```
   ORACLE_HOME="complete-path-to-the-Oracle-Home-Directory"
   export ORACLE_HOME

   ORACLE_SID="Oracle-Net8ConnectionName"
   export ORACLE_SID

   LD_LIBRARY_PATH=complete-path-to-the-Oracle-Home-Directory/
   lib:$LD_LIBARY_PATH}

   PRE_CLASSPATH=complete-path-to-the-Oracle-Home-Directory/jdbc/
   lib/classes12.zip:complete-path-to-the-Oracle-Home-Directory/
   jdbc/lib/nls_charset12.zip:$PRE_CLASSPATH
   ```

3. Save and close the `startManagedWebLogic.sh` (Linux or Solaris) or `startManagedWebLogic.cmd` (Windows) file.

Oracle Installations: Identifying the Type 4 JDBC Driver

Complete the following steps to specify the location of the Type 4 driver.

1. Use a text editor to open the `startWebLogic` and `startManagedWebLogic` scripts:
   - On Windows: `startWebLogic.cmd` and `startManagedWebLogic.cmd`
   - On Solaris: `startWebLogic.sh` and `startManagedWebLogic.sh`
2. In both startup scripts, insert the following statement:

   `PRE_CLASSPATH=`*`complete-path-to-the-Oracle-Home-Directory/`*`jdbc/lib/classes12.zip:`*`complete-path-to-the-Oracle-Home-Directory/`*`jdbc/lib/nls_charset12.zip:$PRE_CLASSPATH`
3. Save and close the `startManagedWebLogic.sh` (Linux or Solaris) or `startManagedWebLogic.cmd` (Windows) file.

Restart WebLogic and Verify the Database Connections

After you finsh the JDBC driver configuration, stop and restart WebLogic and then verify that it can connect to the database.

Stop and restart both the WebLogic Admin Server and the Managed Server

When the Managed Server stops writing status messages to the console or shell or log, scroll back and examine the output. You have successfully configured the JDBC driver when you can find output similar to the following:

```
<Aug 1, 2003 2:15:33 PM EDT> <Info> <JDBC> <001068> <Connection
for pool "mypool817" created.>
<Aug 1, 2003 2:15:33 PM EDT> <Info> <JDBC> <001082> <Creating Data
Source named myjndi817 for pool mypool817>
<Aug 1, 2003 2:15:33 PM EDT> <Info> <JDBC> <001070> <Checking
existence of connection pool mypool817 requested by user
principals=[kernel identity = 256023221]>
```

In this example, the connection pool was named “mypool817” and the data source was named “myjndi817.”

Next Step

Proceed to one of the following two chapters, as appropriate for your operating system:

- Chapter 12, “Before Installing Content Server on Windows”
- Chapter 13, “Before Installing Content Server on Solaris or Linux”

Section 5

Content Server

This section explains how to install and configure Content Server. It contains the following chapters:

Chapter 12

Before Installing Content Server on Windows

This chapter explains what you need to do prior to installing Content Server on Windows. If you are installing on Solaris, see Chapter 13, “Before Installing Content Server on Solaris or Linux” instead.

This chapter contains the following sections:

- Prerequisite Components
- Environment Requirements
- Back Up Your web.xml File
- Clusters Only: Create the Shared File System
- Extract the Installation Program
- Next Step

Prerequisite Components

Before installing Content Server on Windows, you must have already installed and configured the following components:

- A version of Windows (and all necessary service packs) that Content Server supports.
- A version of Oracle or SQL Server 2000 that Content Server supports. Refer to Chapter 4, “Installing the DBMS” for instructions. Confirm that the tablespace is empty.
- A version of IIS that Content Server supports. Refer to Chapter 6, “Installing IIS on Windows” for instructions.
- A version of WebLogic that Content Server supports on all the hosts where you plan to install Content Server. Refer to the the “Application Server” section for instructions.
- A JDBC driver that WebLogic and your DBMS support. Refer to Chapter 5, “Configuring the DBMS for Content Server” and Chapter 11, “Configuring the JDBC Driver” for instructions.

Make sure that you are using the versions listed at the following URL:

`http://cswww.fatwire.com/products/ContentServer`

Environment Requirements

Before installing Content Server, you must ensure that the `PATH` environment variable includes the path to a supported version of JDK. Both WebLogic versions 8.1 and 7.1 provide a supported version of JDK, so the easiest way to fulfill this requirement is to include the path to the JDK provided by WebLogic.

For example, the WebLogic-provided JDK in version 8.1 is located here:

```
WL_HOME\jdk141_02\bin
```

You recorded the value of WL_HOME in Table 6, “WebLogic Installation Parameters” on page 24.

To set the `PATH` variable, do the following:

1. Select Start > Control Panel > System.
2. In the **System Properties** dialog box, select the **Environment** tab.
3. Examine the value assigned to the `PATH` variable. If there is no path setting for a version of JDK, prepend the value set for this variable with:

   ```
   WL_HOME\jdk141_02\bin;
   ```
4. Restart the Windows 2000 server.

Back Up Your web.xml File

Caution

The Content Server installation program will overwrite the `web.xml` file used by the Content Server web application. If you have customized your Content Server `web.xml` file, make a copy of it *before* you install or upgrade Content Server so that your customizations are not lost.

Clusters Only: Create the Shared File System

The cluster needs a shared file system to store common system files and for synchronizing cache activities (resultset, disk, and memory).

Note the following:

- The shared file system should reside either on the database host or another, separate host. In other words, it should not reside on any of the application servers in the cluster or on any of the web servers.
- The Content Server user account (`csdbuser`) on every server in the cluster should have the same name and password.

- The Content Server user account on every server in the cluster must have read, write, and delete access to this shared file system.
- Create a subdirectory in the share to use as the synchronization folder. For example, name it `sync`.

Record the following values in Table 10, "WebLogic Cluster Parameters" on page 26:

- The name of the share
- The name of the `sync` folder in the share

Prepare to Extract the Installation Program

Before extracting the Content Server installation program, do the following:

1. Verify that the database is running.
2. Start the web server.
3. Start the admin server and the managed server on this host.

Extract the Installation Program

1. Create a temporary directory.
2. Run the self-extracting file `cs.exe` from the CD to extract the installation files to a temporary directory that you created.
3. Open a DOS prompt.
4. Change to the directory you specified in step 1. For example, if you extracted the files into `c:\temp_cs`, issue the following command:

   ```
   c:\> cd c:\temp_cs\ContentServer
   ```

5. Type the following to invoke the installation batch file (`csinstall.bat`):

   ```
   c:\temp_cs> csintall
   ```

Next Step

Proceed to Chapter 14, "Installing Content Server."

Chapter 13

Before Installing Content Server on Solaris or Linux

This chapter explains what you need to do prior to installing Content Server on Solaris or Linux. If you are installing on Windows, see Chapter 12, "Before Installing Content Server on Windows" instead.

This chapter contains the following sections:

- Prerequisite Components
- Environment Requirements
- Back Up Your web.xml File
- Prepare to Extract the Installation Program
- Extract the Installation Program
- Next Step

Prerequisite Components

Before installing Content Server on Solaris or Linux, you must have already installed and configured the following components:

- A version of Solaris (and all necessary patches) or Linux that Content Server supports.
- A version of Oracle DBMS that Content Server supports. Refer to "Considerations for Installing Oracle" for instructions. Confirm that the tablespace is empty.
- A version of Apache that Content Server supports. Refer to Chapter 7, "Installing Apache on Solaris or Linux" for instructions.
- A version of WebLogic that Content Server supports on all the machines where you plan to install Content Server. Refer to the "Application Server" section for instructions.
- A JDBC driver that WebLogic and Oracle support. Refer to Chapter 5, "Configuring the DBMS for Content Server" and Chapter 11, "Configuring the JDBC Driver" for instructions.

Make sure that you are using the version listed at the following URL:

`http://cswww.fatwire.com/products/ContentServer`

Environment Requirements

Before installing Content Server, you must ensure that your UNIX environment is set up properly. Specifically, you must:

- Verify HOME Directory Permissions
- Set the PATH

Verify HOME Directory Permissions

Follow these steps to ensure that the `HOME` directory has the correct permissions:

1. Log in as the Solaris or Linux user that you used to install WebLogic. Retrieve this information from Table 5, "Who Installed WebLogic?" on page 24.
2. If necessary, change to the `HOME` directory.

   ```
   $ cd
   ```

3. Check the permissions.

   ```
   $ ls -ald .
   ```

   The output from the ls command should show the following permissions:

   ```
   drwxr-xr-x
   ```

4. If the permissions are incorrect, run the following command:

   ```
   $ chmod 755 .
   ```

Set the PATH

The very first directory in the `PATH` environment variable must contain the path to either the JDK that WebLogic provides or to another JDK version that is 1.3.x or higher.

The values for the WebLogic-provided JDK are as follows:

- For WebLogic 6.1: `WL_HOME/bea/jdk1.3.1_03/bin`
- For WebLogic 7.1: `WL_HOME/bea/jdk1.3.1_06/bin`
- For WebLogic 8.1: `WL_HOME/bea/jdk1.4.1_03/bin`

For example, suppose you installed WebLogic 8.1 in a directory called `local/weblogic8`. In this case, you insert the following value at the very beginning of the `PATH` variable:

```
export PATH=/local/weblogic8/bea/jdk1.4.1_03/bin:$PATH
```

You can edit the `PATH` variable on the shell command line or in a shell startup file.

Back Up Your web.xml File

Caution

The Content Server installation program will overwrite the `web.xml` file used by the Content Server web application. If you have customized your Content Server `web.xml` file, make a copy of it *before* you install Content Server so that your customizations are not lost.

Clusters Only: Create the Shared File System

The cluster needs a shared file system to store common system files and for synchronizing cache activities (resultset, disk, and memory).

Note the following:

- The shared file system should reside either on the database host or another, separate host. In other words, it should not reside on any of the application servers in the cluster or on any of the web servers.
- The Content Server user account (`csdbuser`) on every server in the cluster should have the same name and password.
- The Content Server user account on every server in the cluster must have read, write, and delete access to this shared file system.
- Create a subdirectory in the share to use as the synchronization folder. For example, name it `sync`.

Record the following values in Table 10, “WebLogic Cluster Parameters” on page 26:

- The name of the share
- The name of the `sync` folder in the share

Prepare to Extract the Installation Program

Before extracting the Content Server installation program, do the following:

1. Log in as the Solaris or Linux user that you created. Retrieve this information from Table 5, “Who Installed WebLogic?” on page 24.
2. Start the database and listener as the same user who installed Oracle, if they are not already running. Retrieve this information from Table 2, “DBMS Accounts” on page 22.
3. Start the web server.
4. Start the admin server and the managed server on this host. If this is a cluster installation, start the managed server on this host and the admin server on the admin server host.

Extract the Installation Program

To extract the Content Server installation program, do the following:

1. Create a temporary directory into which you will untar the `cs.tar` file:

   ```
   $ mkdir $HOME/temp_cs
   ```

2. Change to this temporary directory:

   ```
   $ cd $HOME/temp_cs
   ```

3. Untar the `cs.tar` file; for example:

   ```
   $ tar -xvf cs.tar
   ```

   Note

   The GNU `tar` utility does not handle long pathnames in the same way as the Solaris `tar` utility. Do not use the GNU tar utility to unbundle the tar file; you must use the Solaris `tar` utility.

4. The tar program creates a `ContentServer` subdirectory of the temporary directory. Change to that subdirectory by typing:

   ```
   $ cd ContentServer
   ```

5. Invoke the `csinstall` program by typing the following command:

   ```
   $ sh csinstall.sh
   ```

Note for Linux Installations

If the Content Server installer does not function correctly—it does not appear, it becomes unresponsive, or something similar—set the following environmental variable in the WebLogic start script and then restart the application server:

```
LD_ASSUME_KERNEL=2.4.3; export LD_ASSUME_KERNEL
```

Next Step

Proceed to Chapter 14, “Installing Content Server.”

Chapter 14

Installing Content Server

This chapter describes how to install and configure Content Server. It contains the following sections:

- Install Content Server
- Verify the Installation
- About the Content Server Documentation
- Next Steps

Install Content Server

This section advises how to complete the Content Server installation windows and begins with the first window. If you have not yet extracted and invoked the installation program, return to the previous chapter and follow the instructions in one of the following sections, as appropriate for your system:

- For Solaris: "Extract the Installation Program" on page 96.
- For Windows: "Extract the Installation Program" on page 91.

After the installation program starts running, it runs identically on all operating systems. You encounter the following windows during installation:

1. Select Products
2. Installation Directory
3. Installation Type
4. Installation Options
5. Application Server
6. Web Server Configuration
7. Content Server Configuration

8. Satellite Server Configuration
9. Web Server Document Root
10. WebLogic Directory
11. WebLogic Parameters
12. WebLogic Configuration
13. Database Configuration
14. Database Users
15. WebLogic Admin Server Integration (1)
16. WebLogic Admin Server Integration (2)
17. Warning: Prerequisites for Install
18. Content Server Properties
19. WebLogic Install Actions
20. Content Server Install
21. Install

Select Products

The Select Products window asks which product(s) you wish to install. Check the **ContentServer V5.0.1** checkbox.

Click **Next**.

Installation Directory

Cluster Notes

For a cluster installation, you must specify the same pathname on *each* machine in the cluster.

In the Installation Directory window, browse to the directory where you want to install Content Server. If you type the directory name, be sure you use the full pathname and not a relative path.

If you installed your web server and application server on the same system, this is the directory that you created for the Content Server documentation root and that you recorded in Table 12, “Content Server Configuration” on page 27.

The installation program creates the specified directory if it does not already exist.

Click **Next**.

Installation Type

Select one of the following installation options:

Installation Type	Select This Option For...
Single Server	A new single server The primary member of a cluster installation.
Cluster Member	A secondary member of a cluster installation
Upgrade	An upgrade of an existing installation.

Record your choice in the `csType` row of Table 12, “Content Server Configuration” on page 27.

Click **Next**.

Installation Options

Cluster Notes

For a cluster installation, you must select the same installation options for each member of a cluster.

Select none or more of the following add-ons:

Option	Explanation
Portal Example	A sample web portal site that illustrates content delivery techniques for page components, page caching, and image serving. The sample site is useful for verifying your installation and configuration.
Deploy Content Server XML Bridge	You use CS-Bridge XML to receive, deliver, process, route, and transform XML documents to and from other enterprise applications over the web. See the *CSEE Developer's Guide* for an overview of this application.
Deploy Content Server XML Bridge Sample	Some sample code code useful in understanding CS-Bridge XML.
Deploy Debug Servlet	A servlet that will help you debug XML code. This is a useful servlet to install on a development system, but is not recommended on a management or delivery system.

Under **Display Properties**, select **Yes**. You want the Property Editor to launch during the installation because it is very likely that you will need to modify Content Server property values.

Record your installation choices in Table 12, “Content Server Configuration” on page 27.

Click **Next** to continue.

Application Server

Select **WebLogic 8.1** or **WebLogic 7.1** from the drop-down list.

Click **Next**.

Web Server Configuration

Supply the following information:

Menu	Your Response
Web Server Address	You recorded this information in the `WebHost` row of Table 4, "Web Server Parameters."
Web Server Port Number	Determine the answer to the question "Are you installing over a secure web server?" and do one of the following: • If the answer is **Yes**, specify the https port number. • If the answer is **No**, specify the http port number. You recorded the port number in the `WebPort` row of Table 4, "Web Server Parameters."
Are you installing over a secure web server?	Do one of the following: • If you want WebLogic to serve Content Server servlets over a secure port, answer **Yes**. • If you want WebLogic to server Content Server servlets over a nonsecure port, answer **No**. If you select **Yes** but have not yet registered your SSL certificate, the Content Server installation will fail. Therefore, you must register your SSL certificate *before* selecting **Yes**.

Click **Next**.

Content Server Configuration

Cluster Notes

For a cluster installation, you should specify the same Content Server user name and password for each member of the cluster.

Supply the following information:

- In the **Username** field, enter the Content Server user name. (The default user name is `ContentServer`.)
- In the **Password** field, enter a password.
- In the **Confirm Password** field, retype the password.

Note that the restrictions to the length of the password depend on the system you are using to manage users.

Record this information in the *csAdminName* and *csAdminPass* rows of Table 12, "Content Server Configuration" on page 27. Click **Next**.

Satellite Server Configuration

Cluster Notes

For a cluster installation, you should specify the same Satellite Server user name and password for each member of the cluster.

Supply the following information for the CS-Satellite instance that will be installed on this host—that is, the co-resident Satellite servlet on this host:

- In the **Username** field, enter the Satellite servlet (CS-Satellite) user name. The default user name is `SatelliteServer`.
- In the **Password** field, enter a password.
- In the **Confirm Password** field, retype the password.

Note that the restrictions to the length of the password depend on the system you are using to manage users.

Record this information in the *SatName* and *SatPass* rows of Table 12, "Content Server Configuration" on page 27.

Click **Next**.

Web Server Document Root

Enter the following values:

- Web server root directory — Accept the pathname that the installer provides. This pathname should be the same as the directory you recorded in the *csDocRoot* row of Table 12, "Content Server Configuration" on page 27.
- Directory for shared upload folders — For **cluster installations only**. Enter the full pathname of the shared file system that was created for this cluster. If you are installing on Windows 2000, specify the complete directory name, including the drive letter. Record the information in the *csShared* row of Table 12, "Content Server Configuration" on page 27.

The installation program creates the specified directories if they do not already exist.

Click **Next**.

WebLogic Directory

Cluster Notes

For a cluster installation, the WebLogic home directory should have the same name and path on each member of the cluster.

Enter the absolute path name to the directory where you installed WebLogic, that is, `WL_HOME`. For example:

```
/local/bea/weblogic800
```

The directory you specify will be under the *`beaRoot`* directory and will, itself, contain `samples` as a subdirectory. You recorded the value of *`beaRoot`* and `WL_HOME` in Table 6, "WebLogic Installation Parameters" on page 24.

Click **Next.**

WebLogic Parameters

Cluster Notes

For a cluster installation, the domain name, the domain path, and the web application name should be the same for each cluster member.

Enter the following values:

- **WebLogic Administrator Domain Name** – The name of the domain that you created while you installed the application server. The default domain name is `domain1`. Retrieve this information from Table 6, "WebLogic Installation Parameters" on page 24.
- **Path for your WebLogic Domain** – The location of the domain. Retrieve this information from Table 6, "WebLogic Installation Parameters" on page 24.
- **WebLogic Web Application Name** – Enter a name for the Content Server application. For example, `ContentServer`. If this is a cluster installation, be sure to use the exact same web application name for each member of the cluster.

  Record this information in Table 6, "WebLogic Installation Parameters" on page 24.

Select **Y** to specify the Content Server application as the default web application for the server.

Click **Next**.

WebLogic Configuration

Specify that you are running WebLogic as a managed server by selecting **Yes**.

Click **Next**.

Database Configuration

Cluster Notes

For a cluster installation, the database and JDBC driver type and JNDI Data Source name should be the same for each cluster member.

Enter the following values:

- **Select the Database you are using**–Select the appropriate kind of database and JDBC driver type from the drop-down list.
- **Enter JNDI Data Source Name**–Enter the data source name that you specified when you created the database source in the WebLogic Console during the procedure "Create the Data Source" on page 81.

  This name must exactly match the name that you entered in the **JNDI Name** field in the **Create a new JDBC Data Source...** form. You recorded this name in Table 9, "WebLogic Content Server Parameters" on page 25.

Click **Next**.

Database Users

Cluster Notes

For a cluster installation, the Content Server database user name and password must be the same for each cluster member.

Enter the name and password for the database account that Content Server uses to interact with the DBMS in the following fields:

- **Enter Privileged (Read/Write) Database Username:** enter the name of the `csdbuser`.
- **Enter Privileged User Password**: enter the password for the `csdbuser`.
- **Verify the Password**: retype the password.

You can retrieve this information from Table 2, "DBMS Accounts" on page 22.

Click **Next**.

WebLogic Admin Server Integration (1)

Provide the following information:

- **WebLogic Host** – Enter the name of the host that is running the WebLogic Admin Server (**not** the managed server) for this system. Note that if you are installing on a secondary cluster member, the Admin Server is most likely not the localhost that you are installing on.

  Retrieve this information from the *`wlAdminHost`* row of Table 7, "WebLogic Admin Server Parameters" on page 24.

- **WebLogic Port** – Enter the port number on which the WebLogic Admin Server is listening. Typically it is port 7001. Retrieve this information from the *`wlAdminPort`* row of Table 7, "WebLogic Admin Server Parameters" on page 24.

Click **Next**.

WebLogic Admin Server Integration (2)

Cluster Notes

For a cluster installation, the WebLogic Server Administrator username and password must be the same for each cluster member.

Provide values for the following fields:

- **WebLogic Server Administrator:** enter the user name of the WebLogic administrator password. You recorded this information in Table 7, "WebLogic Admin Server Parameters" on page 24
- **WebLogic Server Administrator Password:** enter the password for the admin user.
- **Verify Password:** retype the password.

Click **Next**.

Warning: Prerequisites for Install

Review the prerequisites and verify that your system meets them. Then click **Next**.

After verifying that the prerequisites have been met, the installation program begins to install Content Server.

The installation program displays information in the Install window—a status bar that shows the progress of the installation, log messages—and logs messages to the following installation log file:

```
csRoot/omninstallinfo/install_log.log
```

Content Server Properties

Eventually the installation program invokes the Property Editor and displays the `futuretense.ini` file.

Examine the conditions in the following steps and complete those that are appropriate for your system:

1. If you are using the Oracle Type 4 (thin) driver, complete these steps:
   a. Select the **Database** tab.
   b. Change the value of `cc.bigtext` to `VARCHAR (2000)` and click **Apply**.
   c. Select **File** > **Save.**
2. If you are using Oracle version 9.2, complete these steps:
   a. Select the **Database** tab.
   b. Change the value for the `cc.datetime` property to `TIMESTAMP` and click **Apply**.

   c. Select **File** > **Save**.

3. If you are installing Content Server on Windows 2000, your DBMS is SQL Server 2000, and you plan to use the CS-Desktop feature, complete these steps:
   a. Select the **Database** tab.
   b. Change the value of the `cc.char` property to `NCHAR` and click **Apply**.
   c. Change the value of hte `cc.varchar` property to `NVARCHAR` and click **Apply**.
   d. Change the value of the `cc.bigtext` property to `NTEXT` and click **Apply**.
   e. Change the value of the `cc.maxvarchar` property to `4000` and click **Apply**.
   f. Select **File** > **Save**.
4. See Appendix B, "More About Properties" for suggestions about additional properties that you might want to adjust during. For example, to enable Content Server debugging, you would set values for several debugging properties.
5. Select **File** > **Save**, even if you have personally made no changes to any property values.
6. Select **File** > **Exit** to close the Property Editor

   The installation resumes. Evenutually the Content Server installer opens the `jsprefresh.ini` file in the Property Editor.
7. Select **File** > **Exit** to close the Property Editor.

   The installation resumes.

WebLogic Install Actions

When you finish editing properties and close the Property Editor, a second **Prerequisites** window appears.

You **must** complete the following steps before you click OK in this window:

- Restart the Admin Server and Managed Server
- Configure the Web Application that Represents Content Server
- Reduce JSP Recompilation (Optional)
- Stop and Restart the Admin and Managed Servers
- Continue with the Content Server Installation

Do **not** click **OK** on the **Prerequisites** window until all of these tasks are completed.

Restart the Admin Server and Managed Server

If the Admin Server is not running, start it. If it is running, stop it and then restart it.

If the WebLogic managed server is not running, start it. If it is running, stop it and then restart it.

Configure the Web Application that Represents Content Server

You configure the web application differently based on whether you are using WebLogic 8.1 and 7.1. Complete the procedure that is appropriate for your system.

WebLogic 8.1 Steps

Complete the following steps for WebLogic 8.1 installations:

1. Start the WebLogic Admin Console.
2. Expand your domain (`csdomain`) by clicking the plus (+) sign next to your domain name.
3. Expand **Deployments** and then select **Web Applications Modules**.
4. On the **Configuration** tab, select **Deploy a new Web Application Module**.
5. Click **Applications** (not the Select link next to it).
6. In the list, select the radio button next to the name of your Content Server web application and click **Target Module**.
7. In the **Deploy a Web Application Module** form, select the name of the managed server on this host and click **Continue**.
8. In the **Review your choices and deploy** window, under **Source Accessibility**, select **I will make the Web Application module accessible from the following location**.
9. Click **Deploy**.
10. Wait until you see the **Success** message in the **Status** column for the web application, and then continue to "Reduce JSP Recompilation (Optional)" on page 107.

WebLogic 7.1 Steps

For WebLogic 7.1 installations, you create the Content Server web application, delete the one that WebLogic creates (because it modifies the name that you specified during the Content Server installation), and then specify that it is the default web application on this system.

Complete the following steps:

1. Start the WebLogic Admin Console.
2. Expand your domain (`csdomain`) by clicking the plus (+) sign next to your domain name.
3. Expand **Deployments** and then select **Web Applications**.
4. In the **Web Applications** form, click **Configure a New Web Application**.
5. Click **Applications** (not the Select link next to it).
6. In the list, click the **Select** link next to the name of the Content Server web application. (You named this in the installation window named "WebLogic Parameters.")
7. In the **Configure Application or Component** window, move the managed server from the **Available Servers** list box to the **Target Servers** list box. (You logged this name in Table 8, "WebLogic Managed Server Parameters" on page 25.)
8. Click **Configure and Deploy**.
9. Wait until you see the message "Completed" displayed as the status in the Deployment Activity section, then, under the domain node on the tree, select **Deployments** > **Web Applications**.

10. In the **Web Applications** form, delete the web application that WebLogic created by default while the Content Server installation program was running.

    WebLogic alters the web application name that you specified in the "WebLogic Parameters" window. For example, if you named your Content Server web application `ContentServer`, WebLogic creates a web application for it but displays the name as `_appsdir_ContentServer_dir`.

    Delete this web application.

    Note

    It may take more than one try to get WebLogic to delete the web application.

11. In the tree, under the domain (*`csdomain`*), expand **Servers** and select the name of the managed server that you created to run Content Server (*`wlCSName`*) on the system you are currently installing.

    A configuration form appears in the right pane.

12. Under the **Configuration** tab, click the **Deployment** tab, and change the **Staging Mode** to `Nostage`.

13. Click **Apply**.

14. Under the domain name, select **Servers** > ***wlManagedServerName***

15. In the Servers form, click **Connection** and then click the **HTTP** tab.

16. In the HTTP form, enter values for the fields as follows:

Field	Suggested Value
Default Web Application	Select from the drop-down list the name of the Content Server web application that you specified during the Content Server installation (*`wlCSName`*) and that you configured in the previous procedure "Configure the Web Application that Represents Content Server."
All Other Fields	Leave these fields as is or consult BEA's documentation for more information. The value you set for these fields does not affect Content Server.

17. Click **Apply** to save your changes.

Reduce JSP Recompilation (Optional)

If your system will not run JSP elements, skip this step.

If your system will use JSP elements, note that WebLogic deletes all the `.class` files whenever the WebLogic Managed Server is restarted. The system must then recompile JSP source files, which can be time consuming. To avoid recompiling JSP elements, take the following steps:

1. Open the `weblogic.xml` file in a text editor such as vi or Notepad. It is located in the following directory:

beaRoot\user_projects*csdomain*\applications*WebAppName*\WEB-INF\weblogic.xml

2. Scroll down to the <jsp-param> statement for <param-name>workingDir</param-name>.
3. Uncomment this section by deleting the <!-- and --> characters.
4. Save and exit the file.

Stop and Restart the Admin and Managed Servers

Stop and restart both the admin server and the managed server.

Continue with the Content Server Installation

In the **Prerequisite** window, click **OK**. The installation continues.

Content Server Install

When the installation is complete, a message box appears and indicates the outcome. Follow the instructions in the message and then click **OK**.

Install

Back in the main installation window, click **Exit**.

Verify the Installation

After the Content Server installation program has finished, perform the following tasks to ensure the installation was successful:

- Copy the Content Server Document Root to the Web Server
- Verify the Connections
- Verify That Log Files Are Being Created

Do not attempt to install the Content Server applications before you have verified that Content Server was installed correctly—that it can connect to the database, the application server, and so on.

Copy the Content Server Document Root to the Web Server

If your web server and application server are installed on separate hosts, copy the contents of the *csRoot*/futuretense_cs directory to the futuretense_cs directory that you created on the web server host.

Verify the Connections

To verify that Content Server can communicate with all the other components on the system, complete the following steps:

1. Browse to the following URL:

   http://*WebHost*:*WebPort*/servlet/HelloCS

You recorded *WebHost* and *WebPort* in Table 4, "Web Server Parameters" on page 23.

Content Server should display the simple message:

```
Welcome to Content Server
```

If you do not see this message, see "Verify That Log Files Are Being Created" on page 109 for information about accessing the log file.

2. Browse to the following URL:

   http://*WebHost*:*WebPort*/servlet/CatalogManager?ftcmd=pingdb

   Content Server should display the message:

   ```
   Operation complete.
   ```

   If you do not see this message, then one of the following problems has probably occurred:

   - You did not register the JDBC driver.
   - You supplied the wrong DBMS user name/password combination in the `futuretense.ini` property file.
   - The DBMS is not running.
   - The network connection between your browser and WebLogic or between WebLogic and the DBMS is not up.

3. If you installed the Portal example, browse to the following URL to test the XML version:

   http://*WebHost*:*WebPort*/servlet/ContentServer?pagename=OpenMarket/Samples/NewPortal/XML/main

   An online newspaper should appear.

4. If your site develops JSP code, browse to the following URL to test the JSP version of the Portal example:

   http://*WebHost*:*WebPort*/servlet/ContentServer?pagename=OpenMarket/Samples/NewPortal/JSP/main

   An online newspaper should appear. If it does not, you will not be able to create JSP elements or add new users to your Content Server system.

Verify That Log Files Are Being Created

When the debugging option is enabled (`ft.debug=yes`) in the `futuretense.ini` file, Content Server writes errors and messages to the log file, `futuretense.txt`.

The log file is in the `logs` directory under the **WebLogic Product Directory**. For example:

wlRoot\domain*wldomain**wlCSName*\logs

(You recorded the value of *wlRoot* in Table 6, "WebLogic Installation Parameters" on page 24 and the value of *wlCSName* in Table 9, "WebLogic Content Server Parameters" on page 25.) Delete or archive the log file frequently because a large log file can hurt performance.

On a delivery system, we recommend that you disable debugging.

About the Content Server Documentation

You can access the Content Server documentation from two different places:

- e-docs web site
- The Content Server installation kit

Documentation on the Web Site

We maintain a web site that contains the latest CS documentation, located at a URL that uses the following naming convention:

`http://e-docs.fatwire.com/CSEE/5.`*n*`/index.htm`

where *n* identifies the particular release; for example:

`http://e-docs.fatwire.com/CSEE/5.5.0/index.htm`

We recommend that you check the web site regularly to determine whether you have the most current information available. The web site lets you easily download a package containing all the latest documentation to your local site.

Documentation on the Installation Kit

In the top-level directory of the kit, you'll find the following documentation files:

- `DOC501` (in both `.tar` and `.zip` formats) holds all the manuals associated with this release.
- `ReadMe.htm` is an HTML file containing the release notes.

The installation program does **not** install the documentation on your system. To place this documentation on your system, then you must unpack it yourself. To unpack the `DOC501.zip` file, just use Winzip or an unzip utility.

To unpack the `DOC501.tar` file, use the `tar` command with the `-xvf` keys. For example, assuming that you are installing Content Server from a CD, the following command would unpack the documentation to directory `/local/CSEE_Docs_501`:

```
$ tar -xvf DOC501.tar /local/CSEE_DOC_501
```

Setting Up Help

The main Content Server interface (installed with the content applications) holds a help button, a large question mark symbol. When a user clicks the help button, Content Server redirects their browser to the URL stored in the `cs.documentation` property. By default, the `cs.documentation` property contains the URL of the Content Server documentation web site.

To get help from documentation stored locally, just change the value of the `cs.documentation` property to the local URL. In fact, you can change `cs.documentation` to any URL at which Content Server documentation is stored. However, be sure that you check the Content Server web site periodically so you can download any new or revised documents to your local site.

Next Steps

After verifying the Content Server installation, you are ready to install the Content Server applications and additional CS-Satellite applications.

- For information about installing the content applications, see the separate book *Installing the CS Content Applications* and pay attention to the sequence outlined there.

  For example, if you plan to set up one of the user authentication plugins, you must complete those steps before running the installer for the content applications.
- For information about installing CS-Satellite, see the separate `CS-Satellite Installation Guide`.

If you are installing Content Server on a cluster, there will be additional configuration steps to complete after you install the content applications.

Chapter 15

Uninstalling Content Server and Miscellaneous Notes

This chapter describes some special considerations for particular situations:

- Uninstalling Content Server
- Using CS-Engage with WebLogic 6.1

Uninstalling Content Server

This section explains how to uninstall Content Server. Uninstalling Content Server involves removing the following components:

- Content Server
- WebLogic Server
- DBMS

You must completely remove each component.

Content Server

To clean the disk on which Content Server was installed, delete the entire Content Server installation directory and all its subdirectories.

Application Server

To remove WebLogic:

1. Start the WebLogic Console.
2. In the WebLogic Console, select **Applications**.
3. Select **Enterprise Applications**.
4. Select the name of the enterprise application corresponding to Content Server and remove it.

5. In the application server administration console, select **JDBC**.
6. Select **JDBC Resources**.
7. Select the name of your JDBC resource for Content Server and delete it.
8. In the application server administration console, select **Connection Pools**.
9. Select the name of your connection pool for Content Server and delete it.

DBMS

The Content Server install creates many database tables. To uninstall Content Server completely, delete all the tables created by Content Server using your choice of database tools. If you created a seperate DBMS account to hold Content Server information, you can search for all tables owned by the DBMS account.

Using CS-Engage with WebLogic 6.1

In some versions of WebLogic 6.1, there is a problem with Java execution that appears when you are using CS-Engage. The problem was noted with WebLogic 6.1 with service pack 4.

To prevent such problems, upgrade to version 7.1 or 8.1.

If for some reason, you cannot yet upgrade to a current version of WebLogic, add a directive to the `CLASSPATH` definition in the startup script for the server running the Content Server application. If you are running Content Server on the managed server (as described in this book), you edit the managed server startup script (`startManagedWebLogic.sh`). If you are running Content Server on the admin server, you edit that script (`startWebLogic.sh`).

In either case, append the following to the `CLASSPATH` variable:

```
$JAVA_HOME/lib/tools.jar
```

Section 6

Appendices

This section contains the following two appendices:

- Appendix A, “Building the Apache Web Server”
- Appendix B, “More About Properties”

Appendix A

Building the Apache Web Server

This chapter describes how to obtain the source code for the latest version of Apache and then compile the Apache HTTP Server from that code.

This appendix contains the following sections:

- Before You Build Apache from Source
- Building Apache HTTP Server
- Verifying the Build

Before You Build Apache from Source

Before you attempt to build the Apache HTTP Server, make sure the following utilities are installed on your system:

- Perl 5.X, which is used by the Apache `configure` utility.
- An appropriate ANSI C compiler:
  - gcc is required for Linux installations
  - Sun's Workshop Pro C compiler generates faster Apache code than gcc, but is available for Solaris installations only

Additionally, determine where—in which directory—you plan to install the Apache HTTP Server. This directory is referred to as the **Apache Root Directory**.

Which User?

The Apache HTTP Server runs under a user account. Because you install Apache on port 80, you must be logged in as the `root` user of the system when you are installing. During the installation, however, Apache changes the user account that the server will use to the "nobody" account.

Building Apache HTTP Server

For more information on compiling and installing Apache in general, see the Apache Web server documentation at: `http://httpd.apache.org/`

To install the Apache web server for use with Content Server, complete following steps:

1. In a UNIX or Linux shell, create a directory to hold the Apache download file. For example, `/local/apache_sources`.

   $ **mkdir /local/apache_sources**

2. In a browser, go to `http://httpd.apache.org`.
3. On that page, select the appropriate **Download** link.
4. On that page, you will see many Apache releases; select Apache 1.3.37 or 2.0.43. Each release is packaged in several different compression formats (for example, `zip, tar.z, tar.gz`, etc.). Select the latest Apache 2.0.x or 1.3.x release in the compression format you are most comfortable with. For example, select `apache_2.0.43.tar.z.`

   A dialog box appears and asks whether you want to open this file or to save it.

5. Select the **Save** option.

   A file selection box appears.

6. Save the file to the directory you created in Step 1.
7. In a UNIX or Linux shell, change directory to the directory you created in Step 1, for example:

   $ **cd /local/apache_sources**

8. Use the appropriate command(s) to unpack the file you downloaded in Step 5. For example, the example in Step 4 was a `tar.z` file and the appropriate commands for it are as follows:

   $ **uncompress apache_2.0.43.tar.Z**
   $ **tar -xvf apache_2.0.43.tar**

9. Use the following command to change directories to the `apache_2.0.43` subdirectory:

   $ **cd apache_2.0.43**

10. Examine the value assigned to your `PATH` variable:

    $ **echo $PATH**

    - The list of directories returned must contain at least the following directories:

      For Solaris:

      `/usr/ccs/bin:/usr/sbin:/bin`

      For Linux:

      `/usr/bin:/bin`

    - (Solaris only) If you are using the Sun Workshop compiler, the Workshop bin directory must appear before the /user/ccs/bin statement. For example:

      `/opt/SUNWspro/bin:/usr/ccs/bin:/usr/sbin:/bin`

    - If you are using the GNU C compiler, make sure `gcc` is in your `PATH`.

- For both Solaris and Linux, make sure that `perl` is in your `PATH`.

11. Run the following `configure` command. Note that the `--prefix` option requires a directory argument. Set this argument to the target directory on your system where you want Apache to be installed. For example, you want Apache to be installed on `/local/apache`. In this case, enter the following:

    For **Solaris** with Apache **2.x**

    ```
    $ ./configure --prefix=/local/apache --enable-module=so \
    ```

    For **Solaris** with Apache **1.3.x**

    ```
    $ ./configure --prefix=/local/apache --enable-module=so \
    --enable-rule=SHARED_CORE
    ```

    For **Linux** with either Apache 2.x or Apache 1.3.x

    ```
    $ ./configure --prefix=/local/apache --enable-module=so \
    ```

12. Invoke the `make` command with the `clean` argument:

    ```
    # make clean
    ```

13. Invoke the `make` command without any arguments:

    ```
    # make
    ```

14. Invoke the `make` command again, with the install argument:

    ```
    # make install
    ```

    This command outputs a lengthy completion message that starts with:

    ```
    ____________________________________________________________
    You now have successfully built and installed the Apache 2.0
    HTTP Server...
    ```

Verifying the Build

Examine the list of compiled modules in Apache to ensure that the `mod.so.c` module was compiled. To do so, invoke `httpd` with the `-l` option and search for `mod_so` in the output, for example:

```
$ ApacheRoot/bin/httpd -l | grep 'mod_so'
mod_so.c
```

Note

The `-l` option is the lowercase `L` (not the digit `1`).

If `mod_so.c` does not appear in the output, you performed one of the steps incorrectly. Assuming that `mod_so.c` did appear, Apache is now installed. Return to "Document Your Apache Parameters" on page 50.

Appendix B

More About Properties

The Content Server installer invokes the Property Editor during the Content Server installation. You can then edit any property in any of the .ini files.

This appendix summarizes the properties that are most frequently set during installation. For a complete explanation of every property, see the *Content Server Administrator's Guide*.

After you modify properties, select save and then exit from the Property Editor.

If you did not choose for the Property Editor to appear and you need to set property values, you can start the Property Editor manually and change the appropriate values in the appropriate `.ini` file after the installation completes.

Properties That Control Character Encoding

The `cs.contenttype` defines the outgoing character encoding. Content Server Explorer depends on this setting to display data correctly. By default, this property is set as follows, which is the recommended setting:

```
text/html; charset=UTF-8
```

If you want a different character encoding, change this property. For example, if you want the outgoing encoding to be Shift JIS, set this property as follows:

```
text/html; charset=Shift_JIS
```

Properties That Control Debugging

You can optionally turn debugging on during Content Server installation. Doing so might help you identify problems during the installation. However, turning on deubugging might significantly increase the time it takes to install Content Server. So, turn on debugging only if you need to resolve some issue.

The following properties control debugging:

Table 18: Properties Controlling Debugging

Property	Explanation
`ft.debug`	Set to `yes` to enter debug messages in a log.
`ft.dbdebug`	Set to `yes` to enter database errors in the log.
`ft.logsize`	Set to a large number; for example, `100000`.

Table 19: Properties Controlling SQL Server 2000

Property	Set it to This Value
`cc.char`	`NCHAR`
`cc.varchar`	`NVARCHAR`
`cc.maxvarchar`	`4000`
`cc.bigtext`	`NTEXT`

Properties That Control Documentation

The `cs.documentation` property specifies a URL from which users access Content Server documentation. See "About the Content Server Documentation" on page 110 for details.

Index

X