Content Server Enterprise Edition

Version: 5.5.1

Installing Content Server with IBM WebSphere Application Server

Document Revision Date: May 5, 2004

FatWire SOFTWARE

Installing Content Server with IBM WebSphere Application Server
Document Revision Date: May 5, 2004
Product Version: 5.5.1

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

FatWire Headquarters
FatWire Corporation
330 Old Country Road
Suite 207
Mineola, NY 11501

Contents

Section 1: Introduction

1 Installation Overview 11

Using This Guide 11
System Requirements 12
Hardware Requirements 12
Software Requirements 12
Content Server Environments 13
Development System 13
Management System 13
Delivery System 13
LDAP Support. 14
Language Support 14

2 Documenting Your Configuration 17

Web Server Configuration 18
WebSphere Configuration. 21

Section 2: Database

3 Installing Oracle 25

Before Installing Oracle 25
Configuring Oracle 26
General Tips When Using Oracle. 26
Configuring Character Sets 26
Configuring Oracle on Solaris 27
Creating Tablespaces for Content Server, WebSphere, and Session Data. 27
Testing the Database Installation 34

Copying JDBC Drivers to the Application Server Machine 35
Choosing Between Type 2 and Type 4 Oracle JDBC Drivers 35
Configuring Oracle 8i for Content Server and WebSphere 36

4 Installing DB2 37
Installing the DB2 Control Center on Solaris/AIX 37
Installing DB2 Fix Packs 38
Creating DB2 Databases for WebSphere and Session Data 38
Verifying DB2 Instance User Information for Solaris/AIX 39
Verifying Kernel Parameter Requirements for Solaris 39
Configuring Character Sets 39
Installing the DB2 Client on the WebSphere Machine 40
Cataloging the DB2 Node on the Client 41
Specifying the Database to Be Used by DB2 Clients 42
Specifying JDBC Version for the DB2 Clients 43
Testing WebSphere Communication with DB2 45
Configuring DB2 for Content Server Data 46
Creating a User Account for the Content Server Database 46
Starting the DB2 Control Center 46
Creating a DB2 Database for Content Server 47
Setting Performance Parameters 47
Creating a Tablespace Directory for Content Server 48
Creating a Buffer Pool 48
Creating a Tablespace 48
Creating a System Temporary Tablespace 49
Making the Tablespace Accessible to Content Server 49
Granting Database Privileges to the Content Server User 50
Informing the Application Server Machine About the Content Server Database 50
Configuring DB2 for Session Data 51
Creating an Operating System User for the Session Database User 51
Creating a DB2 Database for Session Data 51
Setting Performance Parameters 51
Creating a Tablespace Directory for the Session Database 52
Creating a Buffer Pool 52
Creating a Table Space 53
Creating a System Temporary Table Space 53
Configuring the Session Database to Use the Table Space 53
Granting Database Privileges to the Session User 54
Informing the Application Server Machine About the Session Database 55

5 Installing SQL Server 57
Before Installing SQL Server 57
Installing SQL Server 2000 58
Configuring Character Sets 58
Configuring SQL Server for Content Server, WebSphere, and Session Data 58

Checking Your Database Configuration 60
Downloading the Type 4 JDBC Driver. 61
Configuring the JDBC Driver on SQL Server 62

Section 3: Application Server

6 Installing WebSphere. 65
Before Installing WebSphere 65
Installing Custom Components. 66
Configuring the JDBC Driver for SQL Server 67
Verifying the Application Path 69
Configuring HTTP Server. 70

7 Configuring WebSphere for Content Server 71
Starting the Web Server 72
Starting IBM HTTP Server on Solaris/AIX 72
Starting IBM HTTP Server on Windows NT/2000 72
Starting the WebSphere Console 72
Starting the WebSphere Console on Solaris/AIX and Windows NT/2000 72
Creating the Data Source and Driver 73
Creating the Content Server Data Source. 73
Creating the Sessions Data Source 75
Creating the Content Server Application Server. 78

Section 4: Content Server

8 Installing Content Server. 83
Before Installing Content Server. 83
Before You Install on Solaris/AIX 84
Before You Install on Windows 84
Extracting the Installation Program 84
On Solaris 84
On AIX 85
On Windows NT/2000 86
Selecting Installation Options 86
Select Products 87
Installation Directory 87
Installation Type 87
Installation Options 88
Application Server 88
Web Server Configuration 88

Content Server Configuration. 89
Satellite Server Configuration 89
Web Server Document Root. 89
WebSphere Root Directory 90
Database Configuration 90
Database Users 90
Warning: Prerequisites for Install. 90
Installing the Software 91
Change Properties (Optional) 91
Deploy the ContentServer.ear File 93
Restart WebSphere 93
Complete the Installation 94
Verify the Content Server Installation 94
Troubleshooting the Installation 94
Optional Tasks After Installing Content Server 95
Set Up Security. 95
Enable Logging 95
Overriding Default User Management for LDAP or Windows NT 95
Accessing Content Server Documentation 96
Documentation on the Web Site. 96
Documentation on the Kit 96
Setting Up Help 96
Uninstalling Content Server 97
Content Server 97
Application Server 97
DBMS. 97

9 Installing Content Server on a Cluster. 99
What Is a WebSphere Cluster? 99
WebSphere Web Server Plug-in. 99
WebSphere Runtime. 100
WebSphere Cluster. 100
WebSphere Server Groups and Clones 100
Overview of Cluster Installation Tasks. 101
Content Server Cluster Installation. 102
Pre-Installation Tasks 102
Installing Content Server on the Primary Cluster Member 103
Installing Content Server on Other Cluster Members 108
Post-Installation Tasks 111
Testing the Cluster 111
Additional Content Server Cluster Properties 112

A Pre-Installation Readiness 115
General Logistics 115
Checklist: Before We Arrive. 116

Architecture Overview 117
Development Environment 118
Management Environment 119
Delivery Environment 120
Software 121
Web Server 121
Application Server 121
DBMS 122
JDBC Driver 123

Index 125

Section 1

Introduction

This section provides the information that you will need before you begin your installation.

It contains the following chapters:

- Chapter 1, “Installation Overview”
- Chapter 2, “Documenting Your Configuration”

Chapter 1

Installation Overview

Content Server Enterprise Edition is designed to work with a web server, an application server, and a database. The *Installing Content Server with IBM WebSphere Application Server* guide provides configuration instructions for Content Server and each component product so that they correctly work together.

This chapter includes the following sections:

- Using This Guide
- System Requirements
- Content Server Environments
- LDAP Support
- Language Support

Use the test procedures provided with each web server, database, or application server to be sure that it is working properly before continuing to the next step.

Using This Guide

The *Installing Content Server with IBM WebSphere Application Server* guide provides information for FatWire installation services, FatWire partners, and system administrators who install the complete Content Server Enterprise Edition (CSEE) environment, including web server, application server, and database. It contains instructions for installing Content Server with IBM WebSphere Application Server version 5.

Please call FatWire™ Technical Support before installing the software according to your configuration license, and note the following:

- To ensure a successful installation, please be sure that FatWire™ installation personnel or a qualified system integrator installs this licensed product on your system.
- Please see the Release Notes for updates and pointers to the latest online information:

  `http://e-docs.fatwire.com/CSEE/5.5.1/index.html`

- Content Server software depends on the correct installation of third-party platform software. Be sure that you are using the correct supported version before installing any third-party software. See the product list at http://cswww.fatwire.com/products/ContentServer

To install Content Server, first read this chapter, and then sequentially follow the instructions in the remaining chapters that apply to your installation configuration.

For each system component that you install, see the specific chapter about that component. After you have completed the installation and configuration, be sure to verify that the component is installed correctly.

Note

This guide does not include full installation instructions for related software, such as the application server and web server. For third-party software, it includes implementation guidelines that enable component software to work with Content Server. When you install the various components, first refer to the related vendor documentation, and use the information in this guide as a supplement.

System Requirements

Content Server is a set of application services and tools that provide content management and delivery features. Content Server requires the following:

- A web server that responds to HTTP requests from a client, returns a page from disk, or makes a request to an application server.
- An application server, which runs application code in response to a request by the web server.
- A database server to store content.

Hardware Requirements

Hardware requirements for Content Server are generally consistent with the hardware requirements for your application server. Refer to the documentation that comes with your application server for hardware requirements information.

Software Requirements

For the latest software requirements, check the Content Server product section on the FatWire web site:

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

Content Server Environments

You typically install Content Server and its related components in three different environments:

- Development System
- Management System
- Delivery System

In addition, FatWire also recommends a fourth environment for testing. Alternatively, you can reconfigure your development system for testing.

Development System

In a typical development system, all components except a web server are installed on a single Windows NT machine. Development systems typically have the following characteristics:

- Portability, for one developer, or a small number of developers
- No failover required
- No firewall considerations

Management System

Management systems typically have the following characteristics:

- Shared by one or more developers or content providers
- Resemble a delivery environment
- No firewall considerations
- Incorporate growth considerations
- Content management issues (for example, frequent content changes)

A management system that is in use around the clock, seven days a week (for example, at a newspaper web site), may closely resemble a delivery system.

Delivery System

A delivery, or public, environment has the following characteristics:

- High-volume access
- Delivery of information, with less frequent content changes and requirements for disk caching
- Firewall considerations
- Load balancing
- Failover

The management and delivery systems are illustrated in the following diagram:

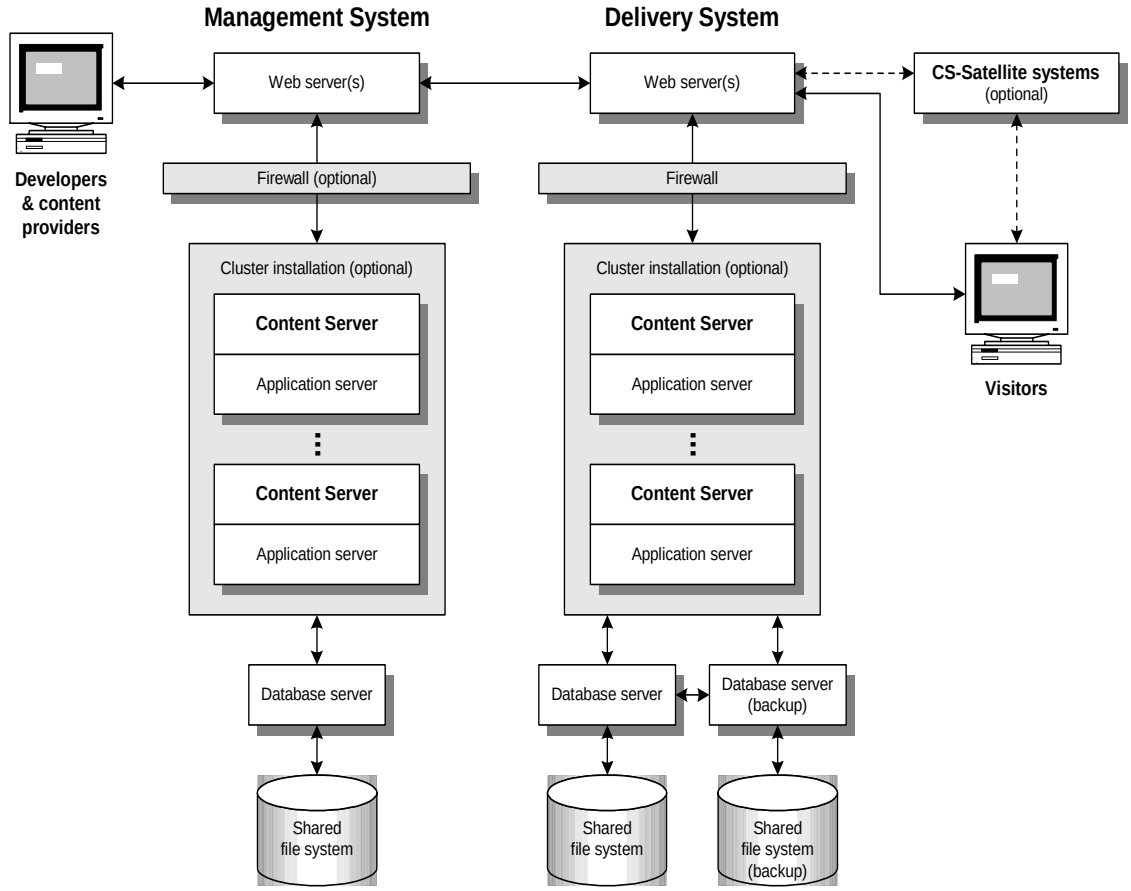


Figure 1: Content Server management and delivery site configurations

LDAP Support

In addition to its native user management capability, Content Server supports the optional LDAP (lightweight directory access protocol) standard. LDAP configuration must be done after Content Server is installed but before you install any Content Server applications. For instructions about configuring LDAP user management with Content Server, refer to *Installing the CSEE Content Applications*.

Language Support

Content Server supports different character sets for the pages it delivers. All commonly used encodings support the basic ASCII character set used for English, plus one or more non-English languages. To use non-ASCII character sets, however, you must configure your Content Server site to handle the right encoding. For more information about

internationalization, including satisfying system requirements, configuring the database, and configuring the application server, refer to the *CSEE Administrator's Guide*.

Chapter 2

Documenting Your Configuration

Performing a Content Server installation requires that you gather information about your database, application server, and Content Server configuration. The following work sheets will help you keep track of this information in one place. To ensure a smooth installation, FatWire recommends that you print the following work sheets and record this information as you perform specific tasks. Taking a little extra time to record your settings during the installation will save you time later when you need to administer the system.

The worksheets are organized in the following sections:

- Web Server Configuration
- WebSphere Configuration

Web Server Configuration

Record information in the appropriate table for either IBM HTTP Server (Solaris, AIX, or Windows) or the Microsoft IIS (Windows only) web server.

Table 1: IBM HTTP Server Configuration

Parameter	Record Values Here
HTTP Server version	
HTTP Server host name	
HTTP Server port number	
User name of install account (administrator user name)	
Password of install account (administrator password)	
HTTP Server root pathname	
How to start HTTP server	
How to stop HTTP server	

Table 2: Microsoft IIS Configuration

Parameter	Record Values Here
IIS version	
IIS host name	
IIS port number	
User name of installer	
Password of installer	
How to start IIS	1. Right-click on **My Computer** icon; select **Manage**. 2. Left-click on the plus sign (+) next to **Internet Information Services**. 3. Right-click on **Default Web Site**; select **Start**.
How to stop IIS	1. Right-click on **My Computer** icon; select **Manage**. 2. Left-click on the plus sign (+) next to **Internet Information Services**. 3. Right-click on **Default Web Site**; select **Stop**.

Database Configuration

Record information in the appropriate table for either the Oracle, IBM DB2, or Microsoft SQL Server databases.

Table 3: Oracle Configuration

Parameter	Record Values Here
Oracle version	
Oracle host name	
Oracle port number	
SID	
User name for Content Server database account	
Password for Content Server database account	
Username for WebSphere administrative database account	
Password for WebSphere administrative database account	
Username for persistent session database account	
Password for persistent session database account	

Table 4: DB2 Configuration

Parameter	Record Values Here
DB2 version	
DB2 host name	
DB2 port number	
DB2 instance	
User name for Content Server database account	
Password for Content Server database account	
User name for WebSphere administrative data account	

Parameter	Record Values Here
Password for WebSphere administrative data account	
Username for persistent session database account	
Password for persistent session database account	

Table 5: SQL Configuration

Parameter	Record Values Here
Oracle version	
Oracle host name	
Oracle port number	
SID	
User name for Content Server database account	
Password for Content Server database account	
Username for WebSphere administrative database account	
Password for WebSphere administrative database account	
Username for persistent session database account	
Password for persistent session database account	

WebSphere Configuration

Table 6: WebSphere Configuration

Parameter	Record Values Here
WebSphere version	
WebSphere host name	
WebSphere Admin Server port number	
WebSphere login name	
WebSphere password	
Name of server representing Content Server	
WebSphere root path name	
Data Source names for Content Server and persistent sessions	For example: `csdata` and `psdata`.
How to start WebSphere Admin Server	`startupServer.sh`
How to start WebSphere Administrative Console	`adminclient.sh`

Section 2

Database

This section describes how to install and configure a DBMS (database management system) and the database client.

It contains the following chapters:

- Chapter 3, “Installing Oracle”
- Chapter 4, “Installing DB2”
- Chapter 5, “Installing SQL Server”

Chapter 3

Installing Oracle

This chapter explains how to configure an Oracle database server for Content Server. It contains the following sections:

- Before Installing Oracle
- Configuring Oracle
- Testing the Database Installation
- Copying JDBC Drivers to the Application Server Machine
- Configuring Oracle 8i for Content Server and WebSphere

Before Installing Oracle

During installation, you can configure Oracle for use by Content Server several ways, the simplest of which is as a single server with a single database on a stand-alone system. Although this configuration may work for a development environment, typical delivery configurations are more complex and require careful consideration for database performance and reliability. If you are installing Oracle on what you intend to be a delivery system, you should rely on the expertise of a database administrator, or at least consult the Oracle installation documentation for the optimal way to install and configure your system.

This document explains how to set up separate databases for Content Server data, WebSphere administrative data, and persistent sessions data. Using separate databases optimizes performance for delivery systems. For development systems, you can use a single database if you want. Understand the configuration you plan to implement at your site before you continue with these instructions because it will affect the database configuration and Content Server cluster options you select later on.

Configuring Oracle

This section includes the following information:

- General Tips When Using Oracle
- Configuring Oracle on Solaris
- Creating Tablespaces for Content Server, WebSphere, and Session Data

Refer to the installation documentation supplied with your database software for specific instructions.

General Tips When Using Oracle

Consider the following when installing an Oracle database for use by Content Server:

- The easiest way to install the database is to select the typical install option and let the Oracle installation software create an initial database with default settings.
- You can create a tablespace for Content Server using SQL*Plus or the Oracle Enterprise Manager. The Enterprise Manager is installed as part of a typical Oracle installation, but it has to be separately configured.
- Be sure to consult the vendor documentation as you install the database software, particularly when performing the following tasks:
  - Checking the `/etc/system` file
  - Creating the Oracle group account
  - Creating Oracle user accounts
  - Checking environment variables

Configuring Character Sets

The database character set must support all characters that you intend to store. Once the database is created, you cannot always easily migrate your data to a different character set. Consider the data to be put in the database beforehand, and configure Oracle according to your requirements.

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

For maximum flexibility, FatWire recommends using UTF-8, which takes up more space in the database but encodes all characters used in modern languages and some archaic languages.

Configuring Oracle on Solaris

After installing Oracle, you need to configure Oracle on your Solaris machine, as follows:

1. Establish Automatic Startup.

   If you want Oracle to start automatically when the system boots, perform the following steps:

   a. Verify that `$ORACLE_SID:$ORACLE_HOME: N|Y` is set to `Y` in `/var/opt/oracle/oratab`.

   b. Create entries (if they do not already exist) similar to the following example. At the end of the `/etc/init.d/dbora` file, be sure to give the full path of the dbstart utility.

   ```
   #/bin/sh
   # Set ORA_HOME to be equivalent to the ORACLE_HOME
   # from which you wish to execute dbstart and
   # dbshut
   # set ORA_OWNER to the user id of the owner of the
   # Oracle datbase in ORA_HOME
   ORA_HOME=/export/home/oracle
   ORA_OWNER=oracle
   if [ ! -f $ORA_HOME/bin/dbstart -o ! -d $ORA_HOME ]
   then
   echo “Oracle startup: cannot start”
   exit
   fi
   case “$1” in
   ‘start’)
   # Start the Oracle databases:
   su - $ORA_OWNER -c $ORA_HOME/bin/dbstart &
   ;;
   esac
   ```

   c. Create soft links to `dbora` by entering:

   ```
   # ln -s /etc/init.d/dbora /etc/rc0.d/K10dbora
   # ln -s /etc/init.d/dbora /etc/rc2.d/S99dbora
   ```

2. Configure the Oracle Database/Client Interface, as described in the vendor documentation.

Creating Tablespaces for Content Server, WebSphere, and Session Data

There are three general steps to perform in preparing an Oracle database for Content Server installation:

1. Create a Default Tablespace for the Content Server database, the WebSphere database, and the persistent sessions database.
2. Create a Temporary Tablespace for the Content Server database, the WebSphere database, and the persistent sessions database.
3. Create User Accounts for the Content Server database, the WebSphere database, and the persistent sessions database.

Note that you can perform these tasks using either the Oracle Enterprise Manager utility or the SQL*Plus command line. Both methods are described here. Depending on your version of Oracle, the Enterprise Manager screens you see may differ slightly from the examples shown in the following sections. The values you need to set remain the same.

Create a Default Tablespace

Using Enterprise Manager, create a default tablespace for Content Server and persistent sessions data, as follows:

1. Start Storage Manager within the Enterprise Manager (`$ORACLE_HOME/bin/oemapp storage`).
2. Log in to the Storage Manager. (If you let Oracle create the database with its defaults, the login is typically **system** and the password is **manager**.)
3. Create a tablespace for the Content Server repository:
   a. In the Storage Manager window, right-click **Tablespace** and select **Create**.

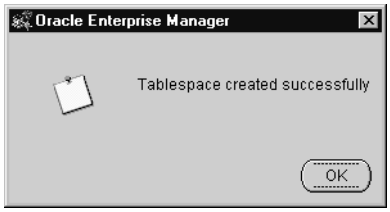
      This displays the Create Tablespace dialog box:

   b. In the **Name** field, enter `csdb.dbf`.
   c. In the **File Name** column, delete the `.ora` extension.
   d. Specify a file size of at least `100 MB`.

      Note: Table size values vary, depending on the size of your hard drive and the amount of space available. The minimum recommended value is 100 MB. You can increase this value, if necessary.

   e. Click **Create**.

f. When the tablespace is created (this may take a minute), Oracle displays the following confirmation:



g. Click **OK**.

4. Create a tablespace for the WebSphere repository. To do this, repeat the same steps you performed to create the Content Server repository:

   a. From the Storage Manager window, right-click **Tablespace** and select **Create**.

   b. In the **Name** field, enter `wsdb.dbf`.

   c. In the **File Name** column, delete the `.ora` extension.

   d. Specify a file size of at least `50 MB`.

      Note: The minimum recommended value is 50 MB. You can increase this value, if necessary.

   e. Click **Create**, and wait for the confirmation dialog box to appear.

   f. From the confirmation dialog box, click **OK**.

5. Create a tablespace for persistent sessions data. To do this, repeat the same steps you performed to create the WebSphere repository:

   a. From the Storage Manager window, right-click **Tablespace** and select **Create**.

   b. In the **Name** field, enter `psdb.dbf`.

   c. In the **File Name** column, delete the `.ora` extension.

   d. Specify a file size of at least `100 MB`.

      Note: The minimum recommended value is 100 MB. You can increase this value, if necessary.

   e. Click **Create**, and wait for the confirmation dialog box to appear.

   f. From the confirmation dialog box, click **OK**.

Creating Default Tablespace Using SQL Commands

Alternatively, you can use the SQL*Plus command line to create default tablespace for the Content Server, WebSphere, and persistent session data repositories. At the SQL prompt, enter the following commands:

```
create tablespace csdb datafile ‘csdb.dbf’ size 100m;
create tablespace wsdb datafile ‘wsdb.dbf’ size 50m;
create tablespace psdb datafile ‘psdb.dbf’ size 100m;
```

Create a Temporary Tablespace

To create a temporary tablespace using Enterprise Manager, repeat the steps in the previous section. For the Content Server database, use the following values:

```
Tablesource name=cstemp
Datafile name=cstemp.dbf
Size=20 MB
```

For the WebSphere administrative database, use these values:

```
Tablesource name=wstemp
Datafile name=wstemp.dbf
Size=20 MB
```

For the session database, use these values:

```
Tablesource name=pstemp
Datafile name=pstemp.dbf
Size=20 MB
```

Creating a Temporary Tablespace Using SQL Commands

Alternatively, you can use the SQL*Plus command line to create a temporary tablespace for the Content Server, WebSphere, and persistent session data repositories. At the SQL prompt, enter the following commands:

```
create temporary tablespace cstemp tempfile ‘cstemp.dbf’ size
20m;
create temporary tablespace wstemp tempfile ‘wstemp.dbf’ size
20m;
create temporary tablespace pstemp tempfile ‘pstemp.dbf’ size
20m;
```

Note: The minimum recommended value is 20 MB. You can increase this value, if necessary.

Create User Accounts

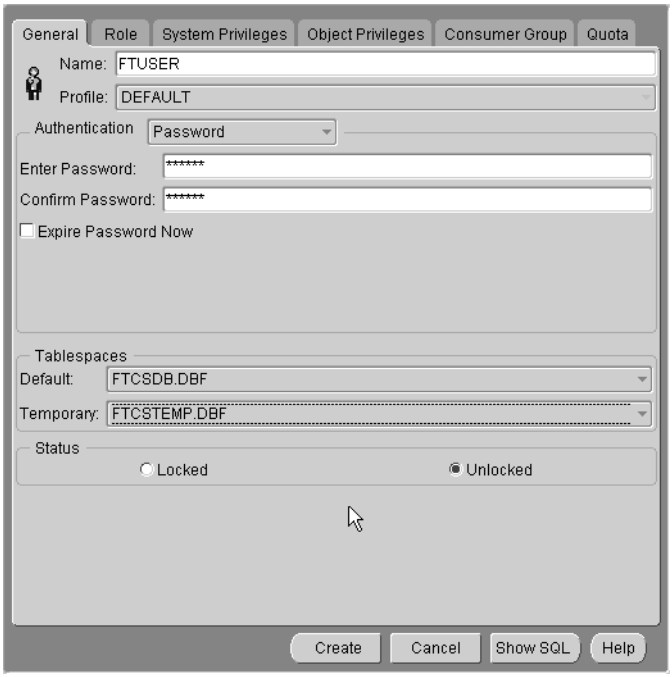
FatWire recommends using separate database accounts for the Content Server, WebSphere, and session data repositories. Note the user name and password for these accounts because you need to specify them during the WebSphere Application Server installation.

To create each account using Enterprise Manager, do the following:

1. Start Security Manager within the Oracle Enterprise Manager (`$ORACLE_HOME/bin/oemapp security`).
2. Log in to the Security Manager. (If you let Oracle create the database with its defaults, the login is typically `system` and the password is `manager`.)

3. Create the user for Content Server data:

   a. In the Security Manager window, right-click **Users** and select **Create**. This displays the Create User dialog box:



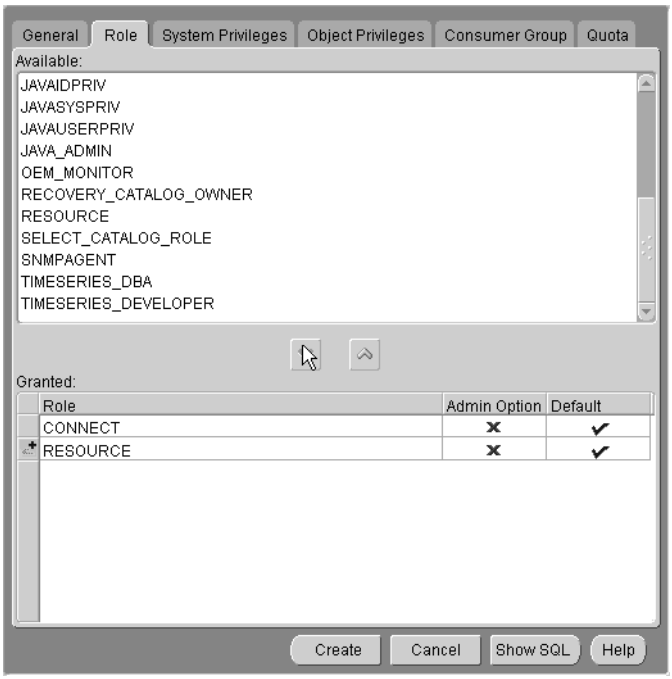
   b. Enter `csuser` in the **Name** field.

   c. Under **Authentication**, enter `csuser` as the password. Enter `csuser` again to confirm the password.

   d. From the drop-down lists under Tablespaces, select **csdb** as the default tablespace, and **cstemp** as the temporary tablespace.

   e. From the **Role** tab, find **CONNECT** and **RESOURCE** in the **Available** list, and click the down-arrow button to add them to the **Granted** list. (**CONNECT** may already be there by default.) Specify these privileges only; do not specify full administrative privileges.

   f. Select the **System Privileges** tab. In the **Available** list, select **Unlimited Tablespace**. Click the down-arrow button to add it to the **Granted** list.



g. Click **Create**.

4. Create the user for WebSphere administrative data:

   a. In the Security Manager window, right-click **Users** and select **Create**. The Create User dialog box appears.

   b. In the **Name** field, enter `ejsadmin`. To avoid a known problem with WebSphere, FatWire recommends using the WebSphere default user `ejsadmin` for installation.

   c. Under **Authentication**, enter `ejsadmin` as the password and confirm it.

   d. From the drop-down lists under Tablespaces, select **wsdb** as the default tablespace, and **wstemp** as the temporary tablespace.

   e. Select the **Role** tab. Find **CONNECT** and **RESOURCE** in the **Available** list and click the down-arrow button to add them to the **Granted** list (CONNECT may already be there by default). Specify these privileges only; do not specify full administrative privileges.

   f. Select the **System Privileges** tab. In the **Available** list, select **Unlimited Tablespace**. Click the down-arrow button to add it to the **Granted** list.

   g. Click **Create**.

5. Create the user for persistent sessions data:

   a. In the Security Manager window, right-click **Users** and select **Create**. The Create User dialog box appears.

   b. In the **Name** field, enter `psuser`.

   c. Under **Authentication**, enter `psuser` as the password and confirm it.

   d. From the drop-down lists under **Tablespaces**, select **psdb** as the default tablespace, and **pstemp** as the temporary tablespace.

e. Select the **Role** tab. Find **CONNECT** and **RESOURCE** in the **Available** list and click the down-arrow button to add them to the **Granted** list (**CONNECT** may already be there by default). Specify these privileges only; do not specify full administrative privileges.

f. Select the **System Privileges** tab. In the **Available** list, select **Unlimited Tablespace**. Click the down-arrow button to add it to the **Granted** list.

g. Click **Create**.

6. Exit the Security Manager.

Creating User Accounts with SQL Commands

Alternatively, you can use SQL commands to create user accounts. To create the Content Server account, enter the following commands at the SQL prompt:

```
create user csuser identified by csuser
default tablespace csdb
quota unlimited on csdb
temporary tablespace cstemp;
grant connect, resource to csuser;
```

To create the WebSphere administrative account, enter the following commands:

```
create user ejsadmin identified by ejsadmin
default tablespace wsdb
quota unlimited on wsdb
temporary tablespace wstemp;
grant connect, resource to ejsadmin;
```

To create the session user account, enter these commands:

```
create user psuser identified by psuser
default tablespace psdb
quota unlimited on psdb
temporary tablespace pstemp;
grant connect, resource to psuser;
```

Testing the Database Installation

Now that you have installed Oracle and created the tablespaces and the associated user account for each, verify that the user accounts can perform the following tasks in the database:

- Add a table in the database
- Add a row to the table
- Drop the table from the database

Perform the following test for each user and database instance you created:

1. Access SQL*Plus.
2. Log in to the instance (`csdb| wsdb | psdb`) as the appropriate user (`csuser | wsuser | psuser`) with the corresponding password (`csuser | wsuser | psuser`).
3. At the SQL prompt, create a simple table called `authors`. For example:

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

   Oracle should create the table successfully.
4. Add a row to the `authors` table:

   ```
   insert into authors values (‘1001’, ‘Smith’);
   ```

   Oracle adds a row to the `authors` table with the specified values.
5. Drop the `authors` table:

   ```
   drop table authors;
   ```

Make sure this test works. If you cannot successfully complete the previous tasks, verify that you can access the database, and check the permissions for the user account before you do anything else. If you are successful, you can go to the next section to install the web and application servers.

Copying JDBC Drivers to the Application Server Machine

If you want to install the WebSphere application server on a different machine from the one on which you installed Oracle, you must copy the JDBC drivers to the machine on which you plan to install the application server.

Note

If you are installing the application server on the machine on which you installed Oracle, you can can skip the instructions in this section.

To copy the JDBC drivers:

1. On the application server machine, create the following directory path:

   ```
   mkdir /OracleDrivers/jdbc/lib
   ```

2. From the Oracle machine, copy the following JDBC driver files to the `/OracleDrivers/jdbc/lib` directory that you just created on the application server machine:

   ```
   classes12.zip
   nls_charset12.zip
   ```

Choosing Between Type 2 and Type 4 Oracle JDBC Drivers

For Oracle, Content Server supports both Type 2 and Type 4 drivers. There are advantages to each type.

Type 2 drivers have the following advantages:

- Allow for longer text (virtually unlimited) to be stored in the DBMS.
- Work with other Oracle tools to perform database load balancing and failover.

Type 4 drivers have the following advantages and limitations:

- More portable and easier to install than Type 2 drivers.
- Storage in the database is limited of 2000 characters. Longer text is referentially stored in the DBMS, with physical storage on disk.
- Some J2EE applications other than Content Server require Type 4 drivers and cannot work with Type 2 drivers.

Configuring Oracle 8i for Content Server and WebSphere

To configure Oracle 8i for a Content Server installation running on the WebSphere application server, perform the following steps:

1. Log in as the Oracle user to the machine where Oracle is installed.
2. Modify the `$ORACLE_HOME/dbs/init<your_SID>.ora` file so that it includes the following settings:

   ```
   open_cursors=300
   processes=100
   shared_pool_size=9000000
   ```

   Note: These settings are required to support Content Server on the live site on your delivery system. You may need to adjust these settings depending on the activity level of your site. For example, if your site handles very high volume, you may need to set the following:

   ```
   open_cursors=1000
   processes=200
   ```

3. Restart the database.

Chapter 4

Installing DB2

This chapter explains how to configure a DB2 database for Content Server. It contains the following sections:

- Installing the DB2 Control Center on Solaris/AIX
- Installing DB2 Fix Packs
- Creating DB2 Databases for WebSphere and Session Data
- Verifying DB2 Instance User Information for Solaris/AIX
- Verifying Kernel Parameter Requirements for Solaris
- Configuring Character Sets
- Installing the DB2 Client on the WebSphere Machine
- Configuring DB2 for Content Server Data
- Configuring DB2 for Session Data

Note

If you have little experience installing and configuring database software and you encounter problems, consult your vendor documentation or have a database administrator assist you.

Installing the DB2 Control Center on Solaris/AIX

Instructions in this chapter require the DB2 Control Center. The Control Center, which is automatically installed on Windows NT/2000, must be installed manually on Solaris/AIX.

During DB2 7.2 installation on Solaris/AIX (but before you install any DB2 Fix Packs, and before you install WebSphere), you can install the DB2 Control Center. When the

IBM instructions ask you to ensure that the **DB2 Administration Client** option is enabled, specify the Control Center, as follows:

1. From the **DB2 Administration Client** option, highlight **Customize.**
2. Press the **Return** key. The DB2 Administration Client window appears.
3. From the DB2 Administration Client window, select **Control Center** and highlight **OK.**
4. Press the **Return** key.

Installing DB2 Fix Packs

Apply the DB2 Fix Pack 5. Do not create the DB2 instance until after you install DB2 Fix Pack 5.

Creating DB2 Databases for WebSphere and Session Data

Using IBM instructions, create a separate database to hold WebSphere administrative data. A separate database instance for administrative data is recommended for best performance on delivery systems. On development systems, you can specify the WebSphere administrative database for sessions data if you want.

To make it easier to follow configuration steps later, FatWire recommends naming the WebSphere database instance `wsdb`. Create user `ejsadmin` and accept the default tablespace provided.

Know which Type of system configuration (development or delivery) you intend to install before you continue because it will affect the database and Content Server cluster options you select later on. Optionally, you can also install the sample database, and perform the tests recommended for that database.

Note

Do not install WebSphere or create a database for Content Server or persistent sessions data at this time. The Content Server and persistent session databases require custom tablespace and other configuration. Later on in this chapter, you will create the databases required for Content Server and persistent session data.

Creating Database Users

The installation instructions in this book use the default database user `ejsadmin` for the WebSphere administrative data repository. If you use a different user name, be sure that you substitute that name for `ejsadmin` where required in these instructions. If you use a name other than `ejsadmin`, you must also change the user name for the `dbSchema` property in the WebSphere `admin.config` file. (This is a known problem with WebSphere.)

Verifying DB2 Instance User Information for Solaris/AIX

After you install the DB2 Fix Pack, you create the DB2 instance for that machine. At that time, DB2 creates the following:

- A `db2inst1` operating system user for running the DB2 instance
- `db2asv` and `db2fence1` users
- One or more operating system user groups to which these users belong

Be sure to record user name and password information, particularly for `db2inst1`.

Verifying Kernel Parameter Requirements for Solaris

According to the IBM document *Installing WebSphere Using DB2 on Solaris*, certain files (for example, `kernel.param.512MB` and `kernel.param.256MB`) contain the appropriate kernel parameter changes needed on Solaris for DB2. As a result, these files may be incorrectly configured on your system. Consult the DB2 documentation to verify the correct settings, as follows:

1. From your DB2 installation `/doc/html/` directory, look in the table of contents under *Quick Beginnings*.
2. Find the *Quick Beginnings for DB2 Universal Database for UNIX* document.
3. Read the *Updating Solaris Kernel Configuration Parameters* article.

Configuring Character Sets

The database character set must support all characters that you intend to store. Once the database is created, you cannot always easily migrate your data to a different character set. Consider the data to be put in the database beforehand, and configure DB2 according to your requirements.

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

For maximum flexibility, FatWire recommends using UTF-8, which takes up more space in the database but encodes all characters used in modern languages and some archaic languages.

Installing the DB2 Client on the WebSphere Machine

To install WebSphere on a machine other than the one on which DB2 resides, you must install the DB2 client on the WebSphere machine.

Note

To install WebSphere on a separate AIX machine from DB2, also read the IBM documentation.

To install WebSphere and the DB2 client on a Solaris, AIX, or Windows machine:

1. Log in to the root account or administrator account on the machine on which you want to install WebSphere and the DB2 client.
2. If you run Windows or AIX, go on to the next step. If you run Solaris, set Solaris kernel configuration parameters to support the DB2 client, as follows:
   a. Add the following entries to the `/etc/system` directory on the DB2 client machine:

      ```
      set msgsys:msginfo_msgmax=65535
      set msgsys:msginfo_msgmnb=65535
      set msgsys:msginfo_msgseg=8192
      set msgsys:msginfo_msgssz=16
      ```

   b. Reboot the DB2 client machine.
3. If you run Windows, go to the next step. If you run Solaris/AIX, verify that the `/export/home` directory exists. The DB2 client installation program cannot create the DB2 client instance unless the `/export/home` directory exists.
4. Install the DB2 7.2 client on the application server machine.

   You can install the DB2 client from the same source (CD or download) from which you installed DB2. A client-only DB2 installation CD and download file is also available.

   The DB2 client installation program tries to create a DB2 client instance and an operating system user and group to run the instance.

   Be sure that you have set the Solaris kernel configuration parameters and rebooted the machine. Otherwise, the client instance will not be created. (See “Verifying Kernel Parameter Requirements for Solaris” on page 39.)
5. Install the DB2 Fix Pack on the DB2 client.

   Install the same DB2 Fix Pack on the DB2 client machine that you installed on the DB2 database machine. (The client-only DB2 installation CD and download file may already include the Fix Pack.)

Cataloging the DB2 Node on the Client

This section provides command-line instructions for configuring a DB2 client to make it aware of the DB2 server instance. If you prefer, you can do the same configuration using the DB2 Control Center.

If you plan to install the WebSphere application server on a different machine from the one on which you installed DB2, you must perform the following steps on each machine that runs a WebSphere application server.

Note

If you are installing the application server on the machine on which you installed DB2, you can skip the instructions in this section.

To tell the DB2 client about the DB2 instance:

1. Create a command prompt on the DB2 client machine that has its environment set up to issue DB2 commands:

   On Solaris/AIX, do one of the following:

   Log in to your DB2 client machine as the DB2 user that you created when you set up the DB2 client instance on that machine. By default, you log in as user `db2inst1`.

   Source the correct environment data to issue DB2 commands. From a Bourne or Korn shell:

   ```
   $ . DB2clientinstalldir/sqllib/db2profile
   ```

   From the C shell:

   ```
   $ source DB2clientinstalldir/sqllib/db2cshrc
   ```

   On Windows NT/2000, choose the following menu items:

   Start > Programs > IBMDB2 > Command Window

2. From the DB2 host, determine the port or service from which the DB2 instance listens:

   Note: Your DB2 host name is the name (or IP address) of the full (not client) DB2 installation machine. The DB2 node name is how the DB2 client identifies the DB2 full installation machine. If the DB2 host name is eight characters or fewer, consider using the host name for the DB2 node name. Otherwise, you should create an eight-character version of the DB2 host name for the DB2 node name.

   For Solaris/AIX:

   a. In the `/etc/services` file on your DB2 machine, look for the entry that defines the port number to which the DB2 Instance listens. Typically, this is port `50000`. For example:

      ```
      db2cdb2inst1 50000/tcp # Connection port for # DB2 instance
      db2inst1
      ```

b. Choose either the service name (for example, `db2dcb2inst1`) or the port number (for example, `50000`) to use for the *`DB2_port_or_servicename`*.

For Windows NT/2000:

By default, a DB2 server machine that runs on Windows NT/2000 uses port 50000. Enter `50000` for the *`DB2_port_or_servicename`*.

3. From the command prompt on the DB2 client machine, enter the following command. (Substitute the appropriate information for your system.)

```
db2 catalog tcpip node DB2nodename remote DB2hostname server
DB2 port_or_servicename
```

4. Verify that the DB2 client can access the DB2 instance you just cataloged. From the command prompt on the DB2 client machine, enter the following command:

```
db2 list node directory
```

Specifying the Database to Be Used by DB2 Clients

This section provides command-line instructions for configuring a DB2 client to recognize the DB2 database instances required to run Content Server on WebSphere. Configuration consists of cataloging databases for WebSphere Application Server, Content Server, and persistent sessions data on each machine that runs WebSphere, other than the one on which you installed DB2. Alternatively, you can catalog the databases using the DB2 Control Center instead of from the command line.

At this time, you should have created a database named `wsdb` to store WebSphere administrative data and optionally created a database named `SAMPLE`.

Note

If you are installing the application server on the machine on which you installed DB2, you can skip the instructions in this section.

To configure the DB2 client to recognize DB2 database instances:

1. From a command prompt on the DB2 Client machine, enter the following command:

```
db2 catalog database database_name as database_alias at node
DB2_nodename
```

Where:

`database_name` is `wsdb` or the name of the database you are describing to this DB2 client.

database_alias is the name that the DB2 Client will use to refer to database *`database_name`* on machine *`DB2_nodename`*. Consider using the *`database_name`* for the *`database_alias`*.

`nodename` is the same node name that you used to define the location where the full DB2 installation with its databases resides.

2. Verify that you just cataloged the database so that this DB2 Client can find it. To do this, enter the following command:

```
db2 list database directory
```

3. To ensure that the DB2 Client can talk to the database, run the following test:

```
db2 connect to database_alias user username using password
db2 "create table testtable1 (testcolumn1 varchar(20))"
db2 "insert into testtable1 values ('testvalue1')"
db2 "select * from testtable1"
db2 drop table testtable1
db2 terminate
```

   Where:

   username and *password* are the user name and password for a user that has permissions to access that database. If user `db2inst1` owns the DB2 server instance, that user should already have access to any instance in the database.

   Quotes (") in the example are only required from the Solaris/AIX command line. The quotes are not needed on a Windows NT/2000 command line, or if you are in the DB2 shell. (To enter a db2 shell, type `db2` at a command line.)

Specifying JDBC Version for the DB2 Clients

Content Server requires the JDBC 2.0 interface. Because DB2 7.1 runs JDBC 1.0 by default, you need to configure each DB2 machine with an application server to use the JDBC 2.0 client. Also, to avoid potential problems, configure the main DB2 machine to use the JDBC 2.0 client, even if you do not plan to run WebSphere on it at this time.

To avoid problems with your DB2 installation, carefully follow the instructions in this section.

Note

For Windows NT/2000, you run a script to specify the JDBC 2.0 driver, but for Solaris/AIX, FatWire recommends that you manually perform this procedure. Although IBM supplies a script to specify JDBC 2.0 for Solaris/AIX, the script does not work for Solaris and, if it fails for AIX, it may delete required JDBC 2.0 files without the possibility of recovering them. For this reason, manually perform these steps as described in the Solaris/AIX section.

Specifying JDBC Version on Solaris/AIX Clients

Perform the following steps on each DB2 client machine that runs an application server, and on the main DB2 server. Instead of using a script file, manually copy the required JDBC 2.0 files to the appropriate directories.

To switch the DB2 installation from JDBC 1.0 to JDBC 2.0:

1. Move to the directory that contains the JDBC 2.0 files:

```
cd DB2InstallDir/java12
```

2. Make backup copies of the files that you are about to replace:

```
cp ../java/db2java.zip ../java/db2java.zip.jdbc10
cp ../lib/libdb2jdbc.so ../lib/libdb2jdbc.so.jdbc10
cp ../bin/db2jd ../bin/db2jd.jdbc10
```

```
cp ../bin/db2jstrt ../bin/db2jstrt.jdbc10
```

3. Copy the JDBC 2.0 files to the directories where they become active:

```
cp db2java.zip ../java
cp libdb2jdbc.so ../lib
cp db2jd ../bin
cp db2jstrt ../bin
```

> **Note**
>
> If you subsequently apply an IBM Fix Pack, or upgrade DB2 to a later version, you will need to determine whether any `/java12` files have been updated. If any files have changed, you will need to repeat these steps and copy them to the `/java` directory.

Specifying JDBC Version on Windows NT/2000 Clients

Perform the following steps on each DB2 client machine that runs an application server, and on the main DB2 server. To switch the DB2 installation from JDBC 1.0 to JDBC 2.0:

1. Stop all DB2 services.

> **Caution**
>
> If you run the script that changes the JDBC interface without stopping all DB2 services, then required file copy operations will fail, resulting in a corrupted JDBC environment.

2. Display the command prompt window:

   Start > Programs > Command Prompt

3. Make a copy of the `/java12` directory and move it to a safe location. FatWire recommends that you create a backup copy in case the script fails.
4. From the command prompt, move to the directory that contains files for the JDBC 2.0 driver:

   `cd \`*`DB2InstallDir`*`\java12`

5. From the command prompt, run the script that copies the JDBC 2.0 files to the directories where they become active:

   `\usejdbc2.bat`

   The script creates a new directory called `\java11,` copies the contents of the `\java` directory to `\java11`, and copies the contents of `\java12` to `\java`. To reflect the active JDBC version, the script changes the `\java12\inuse` text file to read JDBC 2.0.

   If the script fails, you may find that the JDBC 2.0 files in `\java12` have been deleted. Without the files in `\java12`, you can no longer upgrade to JDBC 2.0. Also note that although JDBC 1.0 is still configured, the `\java12\inuse` file may incorrectly read JDBC 2.0. In this case, restore the `\java12` directory with your backup copy and repeat the procedure.

Note

If you subsequently apply an IBM Fix Pack, or upgrade DB2 to a later version, you will need to determine whether any `/java12` files have been updated. If any files have changed, you will need to repeat these steps and copy those files to the `/java` directory.

Testing WebSphere Communication with DB2

To determine whether WebSphere can communicate with DB2, you must test the connection before you continue. Your connection should work if you have correctly done the following:

- Created the WebSphere administrative database
- Installed the DB2 client (if the app server machine is separate from the database machine)
- Supplied WebSphere with a valid database name (or alias)
- Supplied WebSphere with a valid user name (usually `db2inst1`) and password

Follow the test instructions that apply to your operating system.

Testing WebSphere Communication with DB2 on Solaris/ AIX

Before you run *`WebSphereInstallDir`*`/AppServer/bin/startupServer.sh` to start the WebSphere runtime environment, test your connection to the WebSphere administrative database by executing the following `DB2` command from a command line:

```
db2> connect to database_name user username using password
```

For example:

```
db2> connect to WSDB user db2inst1 using db2inst1
```

If you connect successfully, type `terminate` to leave DB2 command-line mode, and go to the next section and configure Content Server data.

If the connection failed, then you have to set some environment information in your *`WebSphereInstallDir`*`/AppServer/bin/startupServer.sh` script before it can work. For the Bourne or Korn shells, put the following line near the start of your `startupServer.sh` script (be sure to include the leading period "."):

```
. $INSTHOME/sqllib/db2profile
```

For the C shell, use a line like the following:

```
source  $INSTHOME/sqllib/db2cshrc
```

Even though you have not yet installed WebSphere, a successful connection indicates that database communication with DB2 is working.

Testing WebSphere Communication with DB2 on Windows

Before you run *`WebSphereInstallDir`*`/AppServer/bin/startupServer.sh` to start the WebSphere runtime environment, test your connection to the WebSphere administrative database by executing the following `DB2` command from a command line:

```
db2> connect to database_name user username using password
```

For example:

```
db2> connect to WSDB user db2inst1 using db2inst1
```

If you connect successfully, type the following command to leave DB2 command-line mode:

```
db2> terminate
```

Otherwise, verify that you have correctly followed the previous database configuration steps. When you have determined that the database is correctly configured, go on to the next section.

Configuring DB2 for Content Server Data

The following instructions explain how to create a DB2 database to hold Content Server data. Perform the steps in this section before you install Content Server.

Caution

Attempting to install Content Server using a DB2 database created without the following configuration instructions will require that you uninstall Content Server, destroy the DB2 database, and create a new DB2 database for Content Server.

Creating a User Account for the Content Server Database

Using your operating system, create a user account to connect DB2 to Content Server. Choose a database user name (FatWire recommends `csuser`), and give the account administrative privileges. Note that the same `db2inst1` user can access all databases.

On the DB2 server, create an operating system user with the name you have chosen (`csuser`, for example). This user does not have to have any special privileges and need not to belong to any special group. Because the user requires an established password, do not choose "**user must change password the next time user logs in**" in the user's profile.

Starting the DB2 Control Center

The steps in this section require you to change settings or issue commands in the DB2 Control Center.

Starting the DB2 Control Center on Solaris/AIX

1. On your DB2 machine, log in as your DB2 instance user, typically `db2inst1`.
2. Issue the command `db2cc` to display the DB2 Control Center.

Starting the DB2 Control Center on Windows NT/2000

1. Log in to your DB2 machine as the user who installed DB2.
2. Choose **Start** > **IBM DB2** > **Control Center**.

Creating a DB2 Database for Content Server

The login account that you use to create the database must have administrative privileges. To create the DB2 database:

1. From the DB2 Control Center, expand the tree view as follows:

   Systems > machine_name > Instances > db2inst1 > Databases
2. Right-click on Databases, and choose **Create > Database Using Wizard**. After a short time, the Create Database Wizard appears.
3. Choose a database name that is eight characters or fewer. For example, `csdb`.
4. Choose a default drive on which to store the data, or keep the default choice.
5. Click **Finish**. (Do not configure database properties from the Advanced Settings tab.)

The name of your new database appears in the tree under the Databases folder. (For example, `csdb`.)

Setting Performance Parameters

The Application heap size, Set Maximum Storage for locklist, and Lock Timeout parameters are typically set too low for Content Server. Adjust them, as follows:

1. From the DB2 Control Center tree:
   a. Right-click **Databases > csdb**.
   b. Choose **Configure . . . .** The Configure Database window appears.
2. From the Configure Database window:
   a. Select the Performance tab.
   b. From the Performance tab, choose the **Application heap size (applheapsz)** parameter. Heap size, which is measured as the number of 4K pages, controls how much space the server side of the JDBC driver has to work with. If the **Value** is currently set to be fewer than `1000`, increase it to `1000` (1000 4K pages).
   c. From the Performance tab, choose the **Set Maximum Storage for locklist (LOCKLIST)** parameter. If the **Value** is set to less than `1000`, increase it to `1000`. If you find that you have problems publishing assets with associations, increase the minimum value as needed.
   d. Click **OK**.
3. From the Configure Database window:
   a. Select the Application tab.
   b. From the Application tab, choose the **Lock Timeout (LOCKTIMEOUT)** parameter, and set the **Value** to `30` (seconds).
   c. Click **OK**.

Creating a Tablespace Directory for Content Server

To create a tablespace directory for the Content Server database:

1. On a command line, or with a file browser, go to where DB2 is installed.
2. Create a subdirectory called `csdb`, for example.
3. Create a subdirectory of that called `cstablespace`. Write down the location of this directory.
4. Create another subdirectory in csdb called `cstemptablespace`. Write down the location of this directory.

For Solaris/AIX, ensure that these two new directories are owned by `db2inst1`, and that they are owned by the same group that owns rest of the directories owned by that user.

Note

DB2 may have problems with the directory names you choose. If you do not follow DB2 conventions for naming directories, DB2 may not accept numbers in directory names.

Creating a Buffer Pool

To create a buffer pool:

1. In the DB2 Control Center tree, right-click on **Databases** > **csdb** > **Buffer Pools** and choose **Create . . . .**
2. Choose a buffer pool name, for example, `PAGESIZE32K`.
3. Choose a page size of 32.
4. Click **OK**. This creates the new buffer pool.
5. Restart the DB2 database for the buffer pool to take effect. For Windows operating systems, do this from the Services dialog.

Creating a Tablespace

To create a tablespace:

1. In the DB2 Control Center tree, right-click on **Databases** > **csdb** > **Table Spaces** and choose **Create** > **Table Space. . .** (Do not select **Table Space Using Wizard...**.)
2. Choose a tablespace name, for example, `cstablespace`.
3. Leave the type of table space set to **Regular**.
4. Click **Add. . .** to display the Add Container dialog box.
5. Choose the `cstablespace` directory created earlier.
6. Click **OK** to close this Add Container dialog box.
7. In the Create Table Space dialog box, click **Advanced** to display advanced properties of this new container.

8. Choose the 32 K buffer pool you created during the previous step.
9. Choose a page size of 32 KB.
10. Click **OK** to close this advanced container properties dialog box.
11. In the Create Table Space dialog box, click **OK**.

Creating a System Temporary Tablespace

To create a system temporary tablespace:

1. In the DB2 Control Center tree, right-click on **Databases > csdb > Table Spaces** and choose **Create > Table Space . . .** (Do not select **Table Space Using Wizard . . .**)
2. Choose a tablespace name, for example, `cstemptablespace`.
3. Choose a Type of table space to System temporary.
4. Click **Add**.... The Add Container dialog box appears.
5. Choose the `cstemptablespace` directory.
6. Click **OK** to close the Add Container dialog box.
7. In the Create Table Space dialog box, click **Advanced**, to display advanced properties of this new container.
8. Choose the buffer pool that you created earlier.
9. Choose a page size of 32 KB.
10. Click **OK** to close this advanced container properties dialog box.
11. In the Create Table Space dialog box, click **OK**.

Making the Tablespace Accessible to Content Server

To enable Content Server to access the tablespace you just created: -

1. In the DB2 Control Center tree, left-click on **Databases > csdb > User** and **Group Objects > DB Groups**.
2. In the right pane, right-click on group **PUBLIC**. Choose **Change**.
3. In the Change Group - PUBLIC dialog box, choose the **Table Space** tab.
4. Click **Add Tablespace.**
5. Choose the `cstablespace` tablespace that you created earlier. Click **OK**.
6. Click on the `cstablespace` tablespace.
7. From the **Privilege: USE** drop-down list at the bottom of the dialog box, choose **Yes**.
8. Click **OK**.

Granting Database Privileges to the Content Server User

To grant the required DB2 privileges to the user for the Content Server database:

1. In the DB2 Control Center tree, right-click on **Databases > csdb > User and Group Objects > DB Users** and choose **Add . . .** to display the Add User dialog box.
2. Click the **Database** tab.
3. From the **User** drop-down menu, select the name of the Content Server database user that you previously added. If the user does not appear in the drop-down menu, create it now.
4. Under **Authorities**, specify options as follows:
   a. Enable (select) the following options:
      Connect database
      Create tables
      Create packages
      Create schemas implicitly
   b. Disable (clear the check boxes for) any remaining options.
5. Click the **Table Space** tab.
6. Click **Add Tablespace . . . .**
7. Choose the `cstablespace` tablespace that you created earlier. Click **OK**.
8. From the **Privilege: USE** drop-down box at the bottom of the dialog box, choose **Grant**.
9. Click **OK** to close the Add User dialog box.

Informing the Application Server Machine About the Content Server Database

Note

Skip this step if you are installing the WebSphere application server on the same machine as DB2.

Follow the instructions in “Specifying the Database to Be Used by DB2 Clients” on page 42. This time, use the name of your new Content Server Database, `csdb`, instead of the name of the WebSphere database, `wsdb`. Repeat this step on each WebSphere machine that uses the new database.

At this point, the `csdb` database is configured for use from your WebSphere machine as user `csuser`, and `csdb` is properly configured to support Content Server. After you configure DB2 for session data, the database will be ready for Content Server installation.

Configuring DB2 for Session Data

The following instructions explain how to create a DB2 database to hold persistent session data. These are the same configuration instructions you used to create the Content Server database.

Creating an Operating System User for the Session Database User

Decide what database user you want to use to connect DB2 to handle Content Server persistent sessions. FatWire recommends `psuser`.

On the DB2 machine, create an operating system user with the name that you have chosen (`psuser`, for example). This user does not require special privileges and need not belong to a special group. Because the user requires an established password, do not choose "**user must change password the next time user logs in**" in the user profile.

Creating a DB2 Database for Session Data

To create a DB2 database for session data:

1. Start the DB2 Control Center.
2. From the DB2 Control Center, expand the tree root as follows:

   Systems > machine_name > Instances > db2inst1 > Databases
3. Right-click on Databases, and choose **Create > Database Using Wizard**. After a short time, the Create Database Wizard appears.
4. Choose a database name that is eight characters or less. For example, `PSDB`.
5. Choose a default drive on which to store the data, or keep the default choice.
6. Click **Finish**. There is no need to choose advanced settings for the various database properties.

After a short time, under the Databases folder in the tree, the name of your new database appears. For example, `psdb`.

Setting Performance Parameters

The Application heap size, Set Maximum Storage for locklist, and Lock Timeout parameters are typically set too low for session data. Adjust them, as follows:

1. From the DB2 Control Center tree:
   a. Right-click **Databases > psdb** .
   b. Choose **Configure . . . .** The Configure Database window appears.
2. From the Configure Database window, select the **Performance** tab.
3. From the **Performance** tab, choose the **Application heap size (applheapsz)** parameter. Heap size, which is measured as the number of 4K pages, controls how much space the server side of the JDBC driver has to work with. If the **Value** is currently set to be fewer than `1000`, increase it to `1000` (1000 4K pages).

4. From the **Performance** tab, choose the **Set Maximum Storage for locklist (LOCKLIST)** parameter. If the **Value** is set to less than `1000`, increase it to `1000`. If you find that you have problems publishing assets with associations, increase the minimum value as needed.
5. Click **OK**.
6. From the Configure Database window, select the **Application** tab.
7. From the **Application** tab, choose the **Lock Timeout (LOCKTIMEOUT)** parameter, and set the **Value** to `30` (seconds).
8. Click **OK**.

Creating a Tablespace Directory for the Session Database

You must create directories for sessions data. For Solaris/AIX, note the following additional configuration requirements:

- Ensure that the newly created directories are owned by user `db2inst1` and that they are owned by the same group that owns other `db2inst1` directories.
- Create the tablespace directories on a different disk from the disks where the Content Server and WebSphere administrative databases reside. Under stress, the disk controller could limit performance for delivery systems. For best performance, FatWire recommends using a separate disk for session data.

To create a tablespace for the session database:

1. On a command line, or with a file browser, move to the location where DB2 is installed.
2. Create a subdirectory, for example, called `psdb`. Be aware that DB2 may not accept numbers in the directory names.
3. Create a subdirectory of the called `psdbtablespace`. Write down the location of this directory.
4. Create another subdirectory in `sessions` called `psdbtemptablespace`. Write down the location of this directory.

Creating a Buffer Pool

To create a buffer pool:

1. In the DB2 Control Center tree, right-click on **Databases** > **psdb** > **Buffer Pools** and choose **Create . . . .**
2. Choose a buffer pool name, for example, `PAGESIZE32K`.
3. Choose a page size of 32.
4. Click **OK**. This creates the new buffer pool.

Creating a Table Space

To create a table space:

1. In the DB2 Control Center tree, right-click on **Databases >psdb > Table Spaces** and choose **Create > Table Space...**. (Do not select **Table Space Using Wizard....**)
2. Choose a tablespace name. For example, `psdbtablespace`.
3. Leave the type of tablespace set to **Regular**.
4. Click **Add . . .** to display the Add Container dialog box.
5. Choose the `psdbtablespace` directory created earlier.
6. Click **OK** to close this Add Container dialog box.
7. On the Create Table Space dialog, click **Advanced** to display advanced properties of this new container.
8. Choose the buffer pool that you created earlier.
9. Choose a page size of 32 KB.
10. Click **OK** to close this advanced container properties dialog box.
11. On the Create Table Space dialog box, click **OK**.

Creating a System Temporary Table Space

To create a system temporary table space:

1. In the DB2 Control Center tree, right-click on **Databases > psdb > Table Spaces** and choose **Create > Table Space.** (Do not select **Table Space Using Wizard . . .**)
2. Choose a tablespace name like `psdbtemptablespace`.
3. Choose a type of tablespace to System temporary.
4. Click **Add**.... The Add Container dialog box appears.
5. Choose the `psdbtemptablespace` directory.
6. Click **OK** to close the Add Container dialog box.
7. In the Create Table Space dialog box, click **Advanced**, to display advanced properties of this new container.
8. Choose the buffer pool that you created earlier.
9. Choose a page size of 32 KB.
10. Click **OK** to close this advanced container properties dialog box.
11. On the Create Table Space dialog box, click **OK**.

Configuring the Session Database to Use the Table Space

To force the session database to use the newly created tablespace:

1. In the DB2 Control Center tree, left-click **Databases > psdb > User** and **Group Objects > DB Groups**.
2. In the right pane, right-click on group **PUBLIC**. Choose **Change**.

3. In the Change Group - PUBLIC dialog box, choose the **Table Space** tab.
4. Click **Add Tablespace.**
5. Choose the `psdbtablespace` tablespace you created earlier. Click **OK**.
6. Click the `psdbtablespace` tablespace.
7. From the **Privilege: USE** drop-down list at the bottom of the dialog box, choose **Yes**.
8. Click on the `USERSPACE1` table space.
9. From the **Privilege: USE** drop-down list at the bottom of the dialog box, choose **No**.
10. Click **OK**.

Granting Database Privileges to the Session User

To grant DB2 privileges to the user for the session database:

1. In the DB2 Control Center tree, right-click on **Databases > psdb > User and Group Objects > DB Users** and choose **Add . . .** to display the Add User dialog box.
2. Click the **Database** tab.
3. From the **User** drop-down list, select the name of the session database user that you added to the operating system earlier. For example, `psuser`.
4. Under **Authorities**, specify options as follows:
   a. Enable (select) the following options:
      Connect database
      Create tables
      Create packages
      Create schemas
   b. Disable (clear check boxes for) the remaining three choices.
5. Click the **Table Space** tab.
6. Click **Add Tablespace**....
7. Choose the `psdbtablespace` tablespace that you created earlier. Click **OK**.
8. From the Privilege: USE drop-down list at the bottom of the dialog box, choose **Grant**.
9. Click **OK** to close the Add User dialog box.

Informing the Application Server Machine About the Session Database

Note

Skip this step if you are installing the WebSphere application server on the same machine as DB2.

Follow the instructions in “Specifying the Database to Be Used by DB2 Clients” on page 42. This time, use the name of your new session database, `psdb`, instead of the name of the WebSphere database `wsdb`. Repeat this step on each WebSphere machine that uses the new database.

The database is now ready for Content Server installation. You can use the `psdb` database from your WebSphere machine as the user `psuser`, and the `psdb` database is properly configured to support Content Server.

Chapter 5

Installing SQL Server

This chapter explains how to configure a Microsoft SQL Server database server for Content Server. It contains the following sections:

- Before Installing SQL Server
- Installing SQL Server 2000
- Configuring Character Sets
- Checking Your Database Configuration
- Downloading the Type 4 JDBC Driver
- Configuring the JDBC Driver on SQL Server

Before Installing SQL Server

During installation, you can configure SQL Server for use by Content Server several ways, the simplest of which is as a single server with a single database on a stand-alone system. Although this configuration may work for a development environment, typical delivery configurations are more complex and require careful consideration for database performance and reliability. If you are installing SQL Server on what you intend to be a delivery system, you should rely on the expertise of a database administrator, or at least consult the SQL Server installation documentation for the optimal way to install and configure your system.

This document explains how to set up separate databases for Content Server data, WebSphere administrative data, and persistent sessions data. Using separate databases optimizes performance for delivery systems. For development systems, you can use a single database if you want. Know the system configuration you need to set up before you continue with these instructions because it will affect the database configuration and Content Server cluster options you select later on.

Installing SQL Server 2000

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

Because only SQL Server Authentication is supported for web applications, you must configure SQL Server to use this authentication mode.

To enable SQL Server authentication:

1. Invoke the SQL Server Enterprise Manager utility.
2. Highlight the SQL Server instance.
3. Change the authentication mode on the Security tab.
4. Restart SQL Server so the change takes effect.

Configuring Character Sets

The database character set must support all characters that you intend to store. Once the database is created, you cannot always easily migrate your data to a different character set. Consider the data to be put in the database beforehand, and configure SQL Server according to your requirements.

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

For maximum flexibility, FatWire recommends using unicode settings, which takes up more space in the database but encodes all characters used in modern languages and some archaic languages. By default, SQL Server uses the UCS-2 character set, which is a double-byte implementation of the unicode standard.

Configuring SQL Server for Content Server, WebSphere, and Session Data

After installing SQL Server, you need to configure SQL Server on your Windows database server. For best performance on delivery systems, create separate databases for WebSphere, persistent sessions, and Content Server data. To be consistent with the examples in this book, FatWire recommends that you name the databases as follows:

Database	Database Name
WebSphere	`wsdb`
Persistent Sessions	`psdb`
Content Server	`csdb`

For each database, follow these steps:

1. Start the SQL Enterprise Manager utility by selecting the following menu options:

   Start > Programs > Microsoft SQL Server > Enterprise Manager

2. From the Enterprise Manager tree view, choose:

   Console Root > Microsoft SQL Servers > *ServerName* **> Databases > New Database**

3. On the **General** tab, enter the name for the database (`wsdb`, `psdb`, or `csdb`).
4. The name you entered for the **General** tab automatically populates the **Data Files** and **Transaction Logs** tabs. Examine these tabs and make any changes, if necessary.
5. Click **OK** to close the window and create the database.
6. From the Enterprise Manager tree view, select **Databases** and verify that the database that you just created (`wsdb`, `psdb`, or `csdb`) appears in the list.
7. Specify login properties, as follows:
   a. From the Enterprise Manager tree, choose:

      Console Root > Microsoft SQL Servers > SQL Server Group > ***ServerName*** **>Security >Logins**

   b. Under **Logins**, right-click on **New Login**. The SQL Server Login Properties dialog box appears.

      Each database (`wsdb`, `psdb`, or `csdb`) requires an associated user account, for which you must specify login properties, as follows:

Database	User Name
wsdb	ejsadmin
psdb	psuser
csdb	csuser

8. From the **General** tab of the Login Properties dialog box, do the following:
   a. For **Name:**, enter `ejsadmin`.
   b. Enable **SQL Server Authentication**. (Click the radio button.)
   c. For **Database:**, select the database (`wsdb`, `psdb`, or `csdb`) that you want to associate with the specified name.
   d. For **Language:**, select the language for your site.
9. On the Roles tab, make no changes. Use the default values.

10. On the Database Access tab, do the following:
    a. Enable the database (`wsdb`, `psdb`, or `csdb`) and associated user name (`ejsadmin`, `psuser`, or `csuser`) combination by selecting the associated check box.
    b. Enable all permissions for the specified database/user combination by selecting the associated check boxes.

       Be sure that you do **not** enable the `db_denydatareader` and `db_denydatawriter` permissions.
    c. Skip the properties configuration (**Properties** button) to accept the default options.
    d. Click **OK**.
11. Repeat steps 1–9 until you have configured all three databases (`wsdb`, `psdb`, and `csdb`).

Checking Your Database Configuration

After installing and configuring the DBMS, you should verify that it works properly. To do so, determine whether you can perform certain tasks while logged in as `csuser`, which is the account that Content Server uses to perform database actions. In particular, Content Server will need to perform the following actions:

- Add a table in the database
- Add a row to the table
- Drop the table from the database

To verify the system, do the following:

1. Access Query Analyzer.
2. Log in to the `csdb` instance. For example, use a login name of `csuser` and a password of `csuser`.
3. At the SQL prompt, create a simple table called `dummy`. For example:

   ```
   SQL) create table dummy (du_id char (11) not null,
   du_lname varchar (40) not null);
   ```

   The DBMS should create the table successfully.
4. Add a row to the `dummy` table:

   ```
   SQL) insert into dummy values ('1001', 'Smith');
   ```

   The DBMS should add a row to the `dummy` table with these values.
5. Drop the `dummy` table:

   ```
   SQL) drop table dummy;
   ```

Make sure this test works. If you can't successfully do any of the previous tasks, check that you can access the database, and check the permissions for the `ftuser` account before you do anything else. If you are successful, proceed to the next section to download a JDBC driver.

Downloading the Type 4 JDBC Driver

Content Server and WebSphere communicate with SQL Server via a JDBC driver. WebSphere 5 requires a Type 4 driver to operate with SQL Server. The JDBC driver enables WebSphere to communicate with an instance of SQL Server.

Note

For delivery systems the actual SQL Server database is installed on a dedicated machine. For management or delivery systems, the application server and database may reside on the same computer.

To download the Type 4 driver for SQL Server:

1. Point your browser to the IBM FTP site. To get the location, consult the readme file that comes with IBM e-Fix PQ62295. Instructions in the file point you to the IBM FTP area that contains the driver.
2. From the IBM FTP area, download the `.jar` files required to configure the application server. The files are contained in a single compressed file called `jdbc2.zip`.
3. Unpack the `jdbc2.zip` file in a temporary directory. The expanded file creates the `\lib` and `\SQLServerJTA` directories.
4. Verify that the `\SQLServerJTA` directory is empty.
5. Verify that the`\lib` directory contains the following `.jar` files:

   - `base.jar`
   - `spy.jar`
   - `sqlserver.jar`
   - `util.jar`

   Remember the location of these files. After you install WebSphere, you will need to copy them to a location on the application server. This enables you to configure WebSphere to use the driver. See “Configuring the JDBC Driver for SQL Server” on page 67.
6. From the same IBM FTP area, download the DataDirect driver. The files are contained in a single compressed file called `SQLServerJTA.zip`.
7. Unpack the `SQLServerJTA.zip` file in a temporary directory. The expanded file creates the `\SQLServerJTA` directory.
8. Verify that the `\SQLServerJTA` directory contains the following files:

   - `instjdbc.sql`
   - `sqljdbc.dll`

   These files will be used to configure the SQL Server installation to use the Type 4 driver. To use the Type 4 driver, you must also install the SQL Server client on the machine that hosts the application server.

Configuring the JDBC Driver on SQL Server

This section explains how to install and configure the JDBC XA stored procedures, which you downloaded from the IBM support site, on SQL Server.

To configure the JDBC driver on SQL Server:

1. Start the SQL Server Distributed Transaction Coordinator process. The SQL Server Distributed Transaction Coordinator process must be running to install the driver. Complete these steps:
   a. Display the SQL Server Enterprise Manager tree view.
   b. From the **Console Root**, select **Microsoft SQL Servers.**
   c. Select your server name **(Windows NT).**
   d. Select **Distributed Transaction Coordinator.**
2. From the temporary directory where you unpacked the `SQLServerJTA.zip` file, copy the `sqljdbc.dll` file to the *`SQL_Server_root`*`\binn directory.` For example:

   ```
   C:\Program Files\Microsoft SQL Server\MSSQL\Binn
   ```
3. From the same temporary directory where you unpacked the `SQLServerJTA.zip` file, copy the `instjdbc.sql` file to any local directory on the database machine.
4. Start the ISQL utility.
5. From the ISQL command prompt, run the `instjdbc.sql` script. Use the following command syntax:

   `>ISQL  -S server -U user -P password -i instjdbc.sql`

   Substitute information for your site as follows:

   `sa` is the SQL Server system administrator user id.

   `sa_password` is the password for the system administrator.

   `server_name` is the name of the server on which SQL Server is running.

   `location` is the full path to the `instjdbc.sql` script.

   For example:

   `>ISQL` **`Usa Pmypasswd S Ojai i c:\instjdbc.sql`**
6. Review the screen output from the `instjdbc.sql` script. Although you can ignore most of the output, look for messages that indicate installation or program-execution problems. The last message should confirm that `instjdbc.sql` ran successfully.

Section 3

Application Server

This section explains how to install and configure the IBM WebSphere Application Server.

It contains the following chapters:

- Chapter 7, “Installing WebSphere”
- Chapter 8, “Configuring WebSphere for Content Server”

Chapter 7

Installing WebSphere

This chapter provides guidelines for installing the IBM WebSphere Application Server software to run Content Server on Sun Solaris, IBM AIX, and Windows NT/2000. It contains the following sections:

- Before Installing WebSphere
- Installing Custom Components
- Configuring the JDBC Driver for SQL Server
- Verifying the Application Path
- Configuring HTTP Server

To ensure that you satisfy prerequisite steps, follow these procedures in the order that they are presented.

Before Installing WebSphere

Before installing WebSphere, you must have installed and configured the following software:

- Oracle, DB2, or SQL Server database
- JDBC driver

IBM HTTP Server, which is the web server included with WebSphere, can be installed as part of the application server installation. The Microsoft IIS web server must be installed separately according to vendor instructions.

Online documentation for IBM WebSphere is available at the IBM WebSphere InfoCenter:

`http://www.ibm.com/software/webservers/appserv/doc/v40/ae/infocenter/`

For helpful supplementary information, also refer to the IBM Redbook for WebSphere 5:

```
http://publib-b.boulder.ibm.com/Redbooks.nsf/
RedpieceAbstracts/sg246176.html
```

Installing Custom Components

A custom installation enables you to specify your database and web server configuration. Choose custom installation options as follows:

1. Invoke the WebSphere installation procedure, and choose **Custom Installation**:
2. From the welcome screen, click **Next**.
1. From the installation menu, choose the components you want to install. By default, all options except the OLT debugger are selected. Specify components as follows:
   a. Select the directory where you want to install the application server and the Web server.
   b. Select **Administration**. This installs the WebSphere administration console.
   c. Select **Application Assembly and Deployment Tools**. This enables the Content Server installation program to register servlets contained in the `contentserver.ear` file with WebSphere.
   d. Select **IBM HTTP Server** (unless you prefer to use Microsoft IIS as your web server). HTTP Server is supported on Content Server running the Solaris, AIX, and Windows systems. IIS is supported for Windows platforms only. If you want to use IIS, you must install it separately according to instructions supplied by Microsoft. After IIS is installed, you must configure the web-server plug-in according to the last step in this section.
   e. Choose **Web Server Plugins** to indicate that you want to install one or more web server plug-ins for WebSphere.
2. Specify the appropriate plug-in for your web server (either **IBM HTTP Server** or IIS). This installs the plug-in on the WebSphere machine. The plug-in enables the web server to forward incoming requests to WebSphere.

> **Note**
>
> If the web server is installed on the same machine as WebSphere, the installation program automatically configures the web server. If your web server is on a remote machine, you can manually copy and configure the plug-in to the appropriate location. This procedure is described as the last step of this section.

3. If your web server is installed on the another machine, copy the plug-in to that machine and configure the web server so that it can locate the plug-in, as follows:
   a. On the machine on which the application server is installed, move to the directory that contains the web server plug-ins:
   ```
   cd /opt/WebSphere/AppServer/config
   ```

b. Copy the `plugin-cfg.xml` plug-in file to a directory on the machine on which the web server is installed.

c. Install the plug-in according to the required method for your web server.

d. Restart the web server so that the newly installed plug-in is recognized.

Note

Enabling Java Security on WebSphere

Anyone who plans to enable Java security must be aware of the following:

Content Server 5.5.1 provides support for enabling Java 2 security on the IBM platform. Content Server creates and embeds the required `was.policy` file within the `EAR` file for deployment.

The `was.policy` file contains a set of permissions that the CS product stack requires in order to operate correctly. The `was.policy` file also guarantees the permissions required by Content Server and its product stack (CS-Direct, CS-Engage, CS-Advantage, and Analysis-Connector). It does create permissions related to third-party products.

The Content Server XML tag `<SYSEXEC>`, which allows template writers to execute commands on the command line, is disabled. Use of this tag will result in `java.security.AccessControlException`. Tag users will have to explicitly add permissions, depending on their intentions.

Content Server continues to create its temporary files under `java.io.tmpdir` rather than the `user.home` directory. It assumes that the user which the app server is running as, has read/ write/delete permissions to the following locations on the disk:

- User `Home` directory and all its subdirectories
- `fatwire` directory and all its subdirectories
- In clustered environments, fatwire shared directories

If you choose to configure Content Server in a manner different from the default, you might have to make adjustments to the `was.policy` file. For example, if the export path does not fall under the `fatwire` directory, you might have to explicitly add that path to the `was.policy` file.

Configuring the JDBC Driver for SQL Server

WebSphere 5 requires additional manual configuration for the SQL Server driver. This section explains how to deploy `.jar` files for the Type-4 (thin) driver, which you previously downloaded from the IBM support site. (See “Downloading the Type 4 JDBC Driver” on page 61.) Thin-driver installation for SQL Server consists of copying `.jar` files to the deployment directory on the application server and editing a configuration file.

Note

You must complete the following procedures before you attempt to configure the data source for the SQL Server driver.

To configure the JDBC driver for SQL Server:

1. Copy driver files for SQL Server driver to the appropriate WebSphere directory.

   From the temporary directory where you unpacked the `connectjdbc2.zip` file, copy the `base.jar`, `spy.jar`, `util.jar` and `sqlserver.jar` files to the *`WebSphereInstallDir`*`\lib` directory. For example:

   `C:> copy *.jar` *`WebSphereInstallDir`*`\lib`

2. Copy the *`WebSphereInstallDir`*`\bin\admin.config` file to a known location. (You must change the `admin.config` file in the next steps—the backup copy ensures that you can start over in case you make a mistake.)

3. Add the `sqlserver.jar` file to the application server class path:

   a. Edit the `admin.config` file.

   b. In the file, locate the following property:

      `com.ibm.ejs.sm.util.process.Nanny.adminServerJvmArgs`

   c. At the end of the paths listed for the property, append the following line:

      `WebSphereInstallDir``/lib/sqlserver.jar`

4. When you selected the Merant database option, the WebSphere installation program generated properties required for the Type 3 driver for SQL Server. Change the property that specifies the driver to reflect the Type 4 driver name:

   a. Edit the `admin.config` file.

   b. In the file, locate the following property:

      `com.ibm.ejs.sm.adminServer.dbdataSourceClassName`

   c. Under the property entry, locate the following value:

      `com.merant.sequelink.jdbcx.datasource.SequeLinkDataSource`

      Change it to this value:

      `com.ibm.websphere.jdbcx.sqlserver.SQLServerDataSource`

5. Change the property that specifies the listener port number for the database machine to the actual port used by your SQL Server installation. The default port used by SQL Server is `1433`. Complete these steps:

   a. Edit the `admin.config` file.

   b. In the file, locate the following property:

      `com.ibm.ejs.sm.adminServer.dbportNumber`

   c. Under the property entry, change the value specified by WebSphere (`19996`)to be the default port number used by SQL Server (`1433`).

      Or, if you are using another listener port, specify that port number instead of the default SQL Server port number.

6. Add a required property:
   a. Edit the `admin.config` file.
   b. In the file, locate the following section:
      `# Repository database settings`
   c. After the database properties in this section, add the following entry:
      `com.ibm.ejs.sm.adminServer.dbselectMethod=cursor`
7. Verify that the database user that you specified during WebSphere installation is used as the value for two properties, as follows:
   a. Edit the `admin.config` file.
   b. In the file, locate the following property and note its value:
      `com.ibm.ejs.sm.adminServer.dbuser`
      The `dbuser` value corresponds to the database user that you specified for the `wsdb` WebSphere database. If you followed the recommended instructions, the value for the `dbuser` property is `ejsadmin`.
   c. Verify that the `dbuser` value is used as the value for the following property:
      `com.ibm.ejs.sm.adminServer.dbSchema`
      If the value for `dbSchema` is different, change it to the `dbuser` value (`ejsadmin`).
8. Save the edits to the `admin.config` file.
9. Reboot the system.
10. Start WebSphere:
    a. Display the Component Services dialog box.
    b. From the Component Services dialog box, select **IBM WS AdminServer** from the list. When selected, the service appears highlighted.
    c. Right click on the highlighted service, and choose **Start**.
11. Check the installation log to verify that WebSphere installed properly:
    a. Using a text editor, open the following log file:
       `WebSphereInstallDir``\logs\tracefile`
    b. Scroll to the bottom of the file and look for the following success message:
       `Server_adminServer open for e-business`

Verifying the Application Path

Content Server requires that the web application path `/servlet` be available. If another application server currently uses `/servlet`, do the following before proceeding with the remaining chapters:

1. Remove the existing application server, or modify its path.
2. Stop WebSphere.
3. Restart WebSphere.

Configuring HTTP Server

After you have installed WebSphere and HTTP Server, manually set web server aliases for Content Server as follows:

1. Move to the web server installation directory:

   Solaris:
   `cd /opt/IBMHTTPD/conf`

   AIX:
   `cd /usr/HTPServer/conf/httpd.conf`

2. Using a text editor, edit the `httpd.conf` file. For example:

   ```
   vi httpd.conf
   ```

3. In the script, locate the section of the file that contains the alias definitions.

4. Add automatic indexing and alias definitions (without trailing slashes), as follows:

   ```
   <Directory /Content_Server_install_path/futuretense_cs>
   Options Indexes
   </Directory>

   Alias /futuretense_cs /Content_Server_install_path/
   futuretense_cs

   Alias /Xcelerate /Content_Server_install_path/futuretense_cs/
   Xcelerate
   ```

   Note

   If you include a trailing slash in the alias definition, the alias will work, but users will have to enter the trailing slash in the browser. Without the slash in the definition, HTTP server will redirect the request and add the slash by default.

Chapter 8

Configuring WebSphere for Content Server

This chapter explains how to configure the IBM WebSphere application server to run Content Server on Sun Solaris, IBM AIX, and Microsoft Windows NT/2000. It contains the following sections:

- Starting the Web Server
- Starting the WebSphere Console
- Creating the Data Source and Driver
- Creating the Content Server Application Server

To ensure that you satisfy prerequisite steps, follow these procedures in the order that they are presented.

Starting the Web Server

WebSphere requires a web server to work. Ensure that IBM HTTP Server or Microsoft IIS is running before you start WebSphere. Follow the instructions that apply to your operating system.

Starting IBM HTTP Server on Solaris/AIX

To start IBM HTTP Server:

1. From the UNIX command line, move to the following directory:

   ```
   % cd /WebServerInstallDir/bin
   ```

2. Run the following command:

   ```
   % ./apachectl start
   ```

Starting IBM HTTP Server on Windows NT/2000

To start IBM HTTP Server:

1. Choose **Start** > **Settings** > **Control Panel** > **Services.**
2. Select the service **IBM HTTP Server**.
3. Click the **Start** button if that service is not already running.

Starting the WebSphere Console

Tasks described in this chapter require the WebSphere Console. To start the WebSphere Console, follow the instructions that apply to your operating system.

Starting the WebSphere Console on Solaris/AIX and Windows NT/2000

1. Start WebSphere:

   a. Move to the following directory:

      ```
      % cd /WebSphereInstallDir/bin
      ```

   b. Run the following command:

      ```
      % startupServer.sh server1
      ```

2. Verify that the server is running by viewing the `tracefile` in the `/WebSphereInstallDir/logs/server1/SystemOut.log` directory. The server is ready when the last line of the file states: `server open for e-business`
3. Open the WebSphere Advanced Administrative Console by going to the following URL:

   `http://hostname:gogo/admin`

Creating the Data Source and Driver

To create the data source and driver:

1. Start the WebSphere Administrative Console. See “Starting the WebSphere Console” on page 72.
2. From the WebSphere Administrative Console, choose:

   Resources > **JDBC Provider**

   The JDBC Provider Properties dialog box appears.
3. From the **General** tab:
   a. Click **Resources**.
   b. Click **JDBC Providers**, located on the left-hand side of the screen:
   c. Click **New**.
   d. Select a provider using the drop-down select box located in **General Properties**.

      For Oracle, choose:

      ```
      Oracle JDBC Thin Driver
      Oracle JDBC OCI8 Driver
      ```

      For DB2, choose:

      ```
      DB2 JDBC Provider
      ```

      For SQL Server, enter:

      ```
      DataDirect Type 4 Driver for MS SQL
      ```
4. Name the driver. The classpath should include the driver `.jar` or `.zip` file, as follows:
   - Oracle: `classes12.zip`
   - DB2: `db2java.zip`
   - SQL Server: `sqlserver.jar`

Creating the Content Server Data Source

If a data source has already been created for another database connection, you can reuse it. Otherwise, create the data source, as follows:

1. Navigate to the **JDBC Provider** page and scroll down.
2. Click the **DataSource V4** link.

> **Note**
>
> The WebSphere Application Server 5.0 can only be used with the V4 datasource.

3. The **Datasources** page appears. Click **New**:
4. Specify the data source’s **Name**, **JNDI Name**, and **Description**.
5. Fill in the following additional data:

   Oracle: 1) Specify the Username and Password of the database’s read/write user.

2) Give the SID the same name as the database name.

DB2: 1) Specify the Username and Password of the database's read/write user.

2) Specify the catalogued database name.

SQL Server: 1) Specify the Username and Password of the database's read/write user.

6. Scroll down and click the **Connection Pool** link.
7. Select **Pool Sizes 10-100 (min-max)**.
8. Click **Apply**. Navigate to the **Datasources** page.
9. Click **Custom Properties** and take the following steps:

- If you are running Oracle, add the following **Custom Properties**:
  a. Click **Add** and enter a custom property called `serverName` in the **Name** field. For the **Value** of `serverName`, enter the host name of the database machine. For example: `mymachine.mycompany.com`
  b. Click **Add** and enter a custom property called `portNumber` in the **Name field**. For the **Value** of `portNumber`, enter the port number on which the Oracle database listens. Typically, Oracle installations use the following port number: `1521`
  c. Click **Add** and enter a custom property called `URL` in the **Name field**. For the **Value** of URL, include the following information, substituting the correct information for your installation:

     `jdbc:oracle:thin:@hostname:port_number:db_name`

     Replace *`hostname`* with the value you set for the `serverName` property. Set *`port_number`* to the value of the `portNumber` property. Set *`db_name`* to the database name you previously set in the **Database Name** field.
  d. Click **Add** and enter a custom property called `disable2Phase` in the **Name field**. Set the value of the `disable2Phase` property to the following value: `true`
- If you are running SQL Server, add the following **Custom Properties**:
  a. Click **Add** and enter a custom property called `serverName` in the **Name** field. For the **Value** of `serverName`, enter the host name of the database machine. For example: `mymachine.mycompany.com`
  b. Click **Add** and enter a custom property called `portNumber` in the **Name** field. For the **Value** of `portNumber`, enter the port number on which the SQL Server database listens. Typically, SQL Server installations use the following port number: `1433`
  c. Click **Add** and enter a custom property called `selectMethod` in the **Name** field. For the **Value** of `selectMethod`, enter: `cursor`

     If you use a two-phase commit connection, you must use `cursor`.

     Optionally, you can set the **Value** of `selectMethod` to `direct`. If appropriate for your resultset size, specifying `direct` can improve performance. For more information about the selectMethod property, refer to the driver documentation from DataDirect Technologies.

d. If you do not need a two-phase commit connection, you can optionally create a custom property that specifies one-phase commit. A one-phase commit connection improves performance by eliminating the overhead of two-phase commit synchronization, which reduces resource use (CPU and JVM heap) on the server.

   To turn off two-phase commit, click **Add** and enter a custom property called `disable2Phase` in the **Name** field. For the **Value** of `disable2Phase`, enter: `true` (The default value is `false`.)

- No custom property is necessary for DB2 installations.

10. To complete creation of the Content Server data source, click **Save**.

Creating the Sessions Data Source

Once you have created the Content Server Data Source, you must create the Sessions Data Source. To create the Sessions Data Source, complete the following steps:

1. Using the tree found on the WebSphere Administration Console, navigate to the Session Management screen as follows: **Application Server** > ***MyServer*** > **Web Container** > **Session Management**, where *MyServer* is the name of the server where you have installed Content Server.
2. Click **Distributed Environment Settings**.
3. Select the **Database** radio button, then click the **Database** link.
4. Enter the DSN properties that you will use for sessions.
5. From the **General** tab:
   a. In the **Name** field, enter an appropriate name for the data source. For example: `sessdata`
   b. **I**n the **JNDI Name** field, enter `jdbc/` followed by data source name you entered in the **Name** field. For example, if you entered `sessdata` for the data source name, then enter `jdbc/sessdata` in the **JNDI Name** field.

Note

The same JNDI Name that you enter here is entered again as the JNDI data source name in Chapter 10, “Installing Content Server.”

   c. In the **Database Name** field, enter the name of the database where you plan to store Content Server data. For example, `sessiondb`.
   d. In the **User ID** field, enter the database user ID for your Content Server data. For Oracle, this is the database account name. For DB2, this is the Content Server database user name. For SQL Server, this is the Content Server database user name.
   e. In the **Password** field, enter the password that corresponds to the user in the **User ID** field.

6. If you are running Oracle, also from the **General** tab, add the following **Custom Properties**:
   a. Enter a custom property called `serverName` in the **Name field**. For the **Value** of `serverName`, enter the host name of the database machine. For example: `mymachine.mycompany.com`
   b. Enter a custom property called `portNumber` in the **Name field**. For the **Value** of `portNumber`, enter the port number on which the Oracle database listens. Typically, Oracle installations use the following port number: `1521`
   c. Enter a custom property called `URL` in the **Name field**. For the **Value** of URL, include the following information, substituting the correct information for your installation:

      `jdbc:oracle:thin:@hostname:port_number:db_name`

      Replace *`hostname`* with the value you set for the `serverName` property. Set *`port_number`* to the value of the `portNumber` property. Set *`db_name`* to the database name you previously set in the **Database Name** field.
   d. Enter a custom property called `disable2Phase` in the **Name field**. Set the value of the `disable2Phase` property to the following value: `true`
7. If you are running SQL Server, also from the **General** tab, add the following **Custom Properties**:
   a. Click **Add** and enter a custom property called `serverName` in the **Name** field. For the **Value** of `serverName`, enter the host name of the database machine. For example: `mymachine.mycompany.com`
   b. Click **Add** and enter a custom property called `portNumber` in the **Name** field. For the **Value** of `portNumber`, enter the port number on which the SQL Server database listens. Typically, SQL Server installations use the following port number: `1433`
   c. Click **Add** and enter a custom property called `selectMethod` in the **Name** field. For the **Value** of `selectMethod`, enter: `cursor`

      If you use a two-phase commit connection, you must use `cursor`.

      Optionally, you can set the **Value** of `selectMethod` to `direct`. If appropriate for your result set size, specifying `direct` can improve performance. For more information about the selectMethod property, refer to the driver documentation from DataDirect Technologies.
   d. If you do not need a two-phase commit connection, you can optionally create a custom property that specifies one-phase commit. A one-phase commit connection improves performance by eliminating the overhead of two-phase commit synchronization, which reduces resource use (CPU and JVM heap) on the server.

      To turn off two-phase commit, click **Add** and enter a custom property called `disable2Phase` in the **Name** field. In the **Value** field for `disable2Phase`, enter: `true` (The default value is `false`.)
8. Click **Save** to save these settings.

9. Navigate to the WebContainer screen as follows: **Application Server** > ***MyServer*** > **Web Container**, where *MyServer* is the name of the server where you have installed Content Server.
10. Click the **Custom Tuning Parameters** link.
11. Select the **Frequency of Session Writes** checkbox.
12. Click the **Save** link to save all of your settings to the Master Configuration.
13. Restart the application server. WebSphere will create table in the database named Session to store session information.

Creating the Content Server Application Server

This section describes how to configure the Content Server application server that will run under WebSphere. Steps in this section generically refer to the directory in which you want to install Content Server as */ContentServerInstallDir*. When you see this notation, substitute the directory path for your own installation, for example, `/opt/ContentServer`, or `C:\ContentServer`.

To create the Content Server application server:

1. If you have not already done so, start the WebSphere Advanced Administrative Console. For instructions, see "Starting the WebSphere Console" on page 72.
2. Click the **Application Servers** link, located in the tree.
3. Click **New**. The New Application Server page appears.
4. Fill in the **Name** field to name the Application Server, then click **Next/Finish**.
5. Note the automatically assigned port by navigating as follows: **Application Server** > ***MyServerName*** > **Web Container** > **HTTP Transports**
6. If your configuration requires it, disable SSL.
7. Navigate to the Default Host page as follows: **Environment** > **Virtual Host** > **Default Host**.
8. Click **Host Aliases**, then **New**.
9. Add the WebSphere host name and port number to the list of host aliases.
10. Select **Application Server**, then click the **Application Server Logging and Trace** link.
    a. In the **Standard Output** field, enter:
    `/WebSphereInstallDir/logs/contentserver_stdout.txt`
    b. In the **Standard Error** field, enter:
    `/WebSphereInstallDir/logs/contentserver_stderr.txt`
11. Navigate to the **JVM** tab as follows: **Application Server** > **Server** > **Web Container** > **Process Definition** > **JVM**:
    a. In the System Properties box, click **Add**.
    b. In the **Name** field, enter: `java.library.path`
    c. In the **Value** field, enter: `/ContentServerInstallDir/bin`

    If, while running Content Server, you notice that your `/WebSphereInstallDir/logs/contentserver_stderr.txt` error file contains the following line:

    ```
    java.lang.UnsatisfiedLinkError: no JniLibraryName in java.library.path
    ```

    then you must add an additional directory specification to the `java.library.path` value field.

    The `java.library.path` variable is the equivalent of setting the Windows 2000 `PATH` variable, the Solaris `LD_LIBRARY_PATH` variable, or the AIX `LIBPATH` variable, but `java.library.path` only sets the variable within the scope of Content Server's JVM.

For the value, specify the directory for the JNI (Java Native Interface) library file, *JniLibraryName*.

d. In the System Properties box, click **Add**.

e. In the **Name** field, enter: `HotSpotOption`

f. In the **Value** field, enter: `client`

g. In the **Initial Java Heap Size** field, enter: `128`

h. In the **Maximum Java Heap Size** field, enter: `128`

Depending on your hardware configuration and web site activity, you may need to increase the **Maximum Java Heap Size** to 256 MB or 512 MB.

12. Specify a character-set encoding that supports all types of data to be handled by your system.

The Java Virtual Machine (JVM) picks up the system locale of the WebSphere installation machine and uses the specified system encoding as the default character encoding for the application server. If character-set encoding for your system locale is consistent with the data to be managed by WebSphere, you can use the default settings, and no additional configuration is required for the application server.

If WebSphere must handle character sets other than the default character set for your system, set the encoding for the application server JVM as appropriate. For example, if the system locale is set to `en_US.ISO8859-1` (English in the United States with Latin-1 encoding) and the application server must handle Japanese double-byte characters, you can specify `UTF-8`, which handles both Latin-1 and Japanese characters.

From the **JVM Settings** tab:

a. In the System Properties box, click **Add**.

b. In the **Name** field, enter: `file.encoding`

c. For UTF-8, for example, in the **Value** field enter: `UTF-8`

13. From the **Database** tab:
    a. Choose the Data Source that you set up for Persistent Session data. For DB2, this is separate data source from the data source on which you plan to store Content Server data. For Oracle, this may be the same data source in which you plan to store Content Server data.
    b. Specify the **Username**, **Password**, and **Confirm Password** that you want to use to store persistent session data in the specified data source.
    c. In the **DB2 Row Size field**, specify `4`.
    d. In the **Table Space Name** field, enter `sessions` .
    e. Enable the **Use Multirow Sessions** check box.
14. Click **OK** to close the Session Manager Service dialog box.
15. Click **OK** to begin Content Server application server generation.
16. Click **OK** to dismiss the Completed Successfully dialog box.

Section 4

Content Server

This section explains how to install Content Server

It contains the following chapters:

- Chapter 10, “Installing Content Server”
- Chapter 11, “Installing Content Server on a Cluster”

Chapter 10

Installing Content Server

This chapter explains the steps for installing Content Server on both Solaris/AIX and Windows NT environments. It contains the following sections:

- Before Installing Content Server
- Extracting the Installation Program
- Selecting Installation Options
- Installing the Software
- Troubleshooting the Installation
- Optional Tasks After Installing Content Server
- Accessing Content Server Documentation
- Uninstalling Content Server

Before Installing Content Server

This section helps you determine if you have completed several necessary configuration steps before you install Content Server.

Caution

The Content Server installation program will overwrite the `web.xml` file. If you have customized your `web.xml` file, make a copy of it before you install Content Server so that your changes will not be lost.

Before You Install on Solaris/AIX

Before installing Content Server, you must complete the following tasks:

- Verify that you are running the proper version of Solaris/AIX and have all the necessary Solaris patches.
- Verify that your DBMS is installed correctly. If you installed Oracle on Solaris/AIX, perform the tasks in “Installing Oracle” on page 25.
- Verify that your application server is installed and configured correctly.
- Verify `HOME` directory permissions, as described in the following section.

Verify HOME Directory Permissions

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

1. In a UNIX shell, log in as `csuser`. If necessary, change to the HOME directory:

   ```
   % cd
   ```

2. Check the permissions by entering the following command:

   ```
   % ls -ald .
   ```

   The permissions should be 755, which generates the following output:

   ```
   rwxr-xr-x
   ```

3. If the permissions are not 755, issue the following command:

   ```
   % chmod 755 .
   ```

Before You Install on Windows

Before installing Content Server on Windows, you should have already performed the following tasks:

- Verify that you are running a version of Windows that Content Server supports.
- Verify that your Web server is properly installed and configured.
- Verify that your DBMS is properly installed and configured.
- Verify that your application server is properly installed and configured.
- Verify special application server requirements, which are described later in this chapter.

Extracting the Installation Program

The way you extract the installation program depends on the operating system.

On Solaris

Before you begin, make sure you are logged in with administration privileges as the same user who installed the application server and the web server.

To install Content Server on Solaris, perform the following steps:

1. Start your DBMS and the web server, if they are not already running.
2. Create a temporary directory into which you untar the `cs.tar` file:

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

3. Change to this temporary directory:

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

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

   ```
   $ tar -xvf /cdrom/solaris/cs.tar
   ```

   Note: The GNU `tar` utility does not handle long pathnames in the same way as the Solaris `tar` utility and does not work for Content Server. You must use the Solaris `tar` utility.

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

   ```
   $ cd ContentServer
   ```

6. Ensure that your PATH variable points to Java version 1.2 or later. WebSphere installs JDK 1.2.2 automatically, so the beginning of your path must point to:

   `/WebSphereInstallDir/``java/bin`

7. To start the installation, type:

   ```
   $ sh csinstall.sh
   ```

On AIX

Before you begin, make sure you are logged in with administration privileges as the same user who installed the application server and the web server.

To install Content Server on AIX, perform the following steps:

1. Start your DBMS and the web server, if they are not already running.
2. Create a temporary directory into which you untar the `cs.tar` file:

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

3. Change to this temporary directory:

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

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

   ```
   $ tar -xvf /cdrom/aix/cs.tar
   ```

   Be sure to use the AIX `tar` utility. The GNU `tar` utility does not handle long pathnames in the same way as the AIX `tar` utility and does not work for Content Server.

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

   ```
   $ cd ContentServer
   ```

6. Ensure that your PATH variable points to Java version 1.2 or later. WebSphere installs JDK 1.2.2 automatically, so the beginning of your path must point to:

   */WebSphereInstallDir/*java/sh;*/WebSphereInstallDir/*java/jre/sh

7. To start the installation, type the following:

   $ **sh csinstall.sh**

On Windows NT/2000

Before you install Content Server on Windows, do the following:

1. Run the self-extracting file cs.exe from the CD to extract the installation files to a temporary directory that you specify.
2. Open a DOS Window.
3. Change to the directory that 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**

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

   c:\temp_cs> **csintall**

Selecting Installation Options

After starting, the installation program runs identically on both Solaris and Windows 2000. The installation program displays the following windows as you progress:

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. WebSphere Root Directory
11. Database Configuration
12. Database Users
13. Warning: Prerequisites for Install

The remainder of this section details each of the preceding screens.

Select Products

The **Select Products** window appears, asking which product(s) you wish to install. Put a check in the box next to the only choice, which is:

```
ContentServer V5.5
```

Then, click **Next**.

Installation Directory

In the **Installation Directory** window, you must supply the full path to where Content Server will be installed. Consider the following:

- You must install Content Server on the same machine where the WebSphere Application Server is installed.
- You must enter a full pathname, not a relative pathname. The installation program creates the specified directory if it does not already exist.
- For a cluster installation, this path must be the same on each machine in the cluster.
- The default pathname is a placeholder only. Be sure to enter the appropriate pathname for your installation.

Click **Next** to continue.

Installation Type

The **Installation Type** window asks you to choose one of the following installation types from the drop-down list:

Installation Type	Select This Option For...
Single Server	A new single server or for the primary member of a cluster installation.
Cluster Member	A new installation of a member of an existing cluster.
Upgrade	An upgrade of an existing installation. If the Content Server database tables have already been created, then you must pick **Upgrade**. The Content Server installation process creates these database tables in the middle of the process. It is possible that the database tables are created even though an installation does not complete. For example, suppose you choose **Single Server** but the installation fails after the database tables were created. Therefore, in order to complete the installation, you must pick **Upgrade** on subsequent attempts to install.

After making your selection, click **Next**. If you chose **Cluster Member**, proceed to the Application Server menu.

Installation Options

In the **Installation Options** window, select from the following options:

Table 7: Installation Options

Option	Explanation
Portal Example	A sample web portal site that illustrates content delivery techniques for page components, page caching, and image serving.
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 Product Overview* for an overview.
Deploy Content Server XML Bridge Sample	Some sample 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.

This form also asks if you want to display the Property Editor. The installation creates a property file named `futuretense.ini`. You use the Property Editor to modify `futuretense.ini` and change the default values. You can also start the Property Editor manually and change values after the installation completes.

Select **Yes** for the Property Editor to automatically appear during installation. This allows you to turn debugging on while the installation program runs.

Note

Turning on debugging can significantly increase the time it takes to install Content Server. Turn on debugging only if necessary to resolve some issue.

Click **Next** to continue.

Application Server

The **Application Server** window asks you to select your application server from the drop-down list.

Choose **WebSphere Application Server** and then click **Next**.

Web Server Configuration

In the **Web Server Configuration** window, enter the host name and web server port number of the server on which the web server is installed. Then, click **Yes** or **No** to indicate whether WebSphere will be serving Content Server servlets over a secure port.

Note

Before you select **Yes**, be sure to register your SSL certificate on the web server. If you select **Yes** but have not yet registered your SSL certificate, the installation will fail.

Then click **Next**.

Content Server Configuration

The **Content Server Configuration** window asks you to supply login information for the Content Server administration account.

- Username—The default user name is `ContentServer`. You can accept this default or change it.
- Password—Enter the password and reconfirm it. The restrictions to the length of the password depend on the system you are using to manage users.

Record this information on the work sheet provided in Chapter 2, “Documenting Your Configuration.”

Click **Next** to continue.

Satellite Server Configuration

The **Satellite Server Configuration** window asks you to supply login information for the Satellite Server administration account.

- Username—The default user name is `SatelliteServer`. You can accept this default or change it.
- Password—Enter the password and reconfirm it. The restrictions to the length of the password depend on the system you are using to manage users.

Record this information on the work sheet provided in Chapter 2, “Documenting Your Configuration.”

Click **Next** to continue.

Web Server Document Root

In the **Web Server Document Root** window, you must supply the **full path** to two directories:

- The web root directory. You created this directory when you configured the web server. The web root directory is required to be `futuretense_cs`, which you can map to any other directory.
- A directory for shared upload folders, which is necessary for cluster installations. If you are installing on Windows 2000, you must include the drive and folder when you specify the shared folder.

You can accept the defaults that are provided or enter a new pathname. If you type in a pathname, it must be a full pathname, not a relative pathname. The installation program creates the specified directory if it does not already exist.

Click **Next** to continue.

WebSphere Root Directory

In the **WebSphere Root Directory** window enter the path to the directory where you will install the application:

WebSphereRootDir/InstalledApps/*NodeName*/ContentServer.ear/CS.war/jsp

where *WebSphereRootDir* is the directory where you have installed the WebSphere Application Server, and *NodeName* is the node name for the WebSphere Application Server.

IMPORTANT! Later in the installation process, when importing the `.ear` file into the WebSphere console, you will be asked to enter an application name. This application name **must** match the *ApplicationName* you entered here.

For example, if you are installing on Solaris and have entered the following as your WebSphere Root Directory:

```
/export/home/WebSphere/AppServer/InstalledApps/
ContentServer.ear/cs.war
```

The application name you enter when importing the `.ear` file must be `ContentServer`.

Database Configuration

In the **Database Configuration** window, enter the appropriate values for each field:

- **Select the Database you are using** — Select the appropriate kind of database and JDBC driver type from the drop-down list.
- **JNDI Data Source Name** – This is the name used for storing database configuration information in the registry. This can be any name you prefer, but it must match the JNDI data source name that you devised when configuring your application server plugin. For example: `jdbc/MYJNDI`.

(Refer to the information you recorded in Chapter 2, “Documenting Your Configuration.”)

Click **Next** to continue.

Database Users

You must supply the name (for example, `csuser`) and password for the Oracle account that Content Server will use to interact with Oracle. You created the name and password when you configured Oracle.

Click **Next** to continue.

Warning: Prerequisites for Install

Verify the prerequisites displayed in the **Prerequisites for Install** window.

Click **Next** to continue.

Installing the Software

After running through all the installation screens, the installation program begins to install Content Server. The installation program displays a status bar showing installation progress, displays log messages in the Install window, and also logs these messages to the following installation log file:

```
$HOME/omninstallinfo/install_log.log
```

After verifying the prerequisites, the installer program begins to install Content Server. You will be directed to do the following tasks during this install procedure:

- Change Properties (Optional)
- Deploy the ContentServer.ear File
- Complete the Installation

Change Properties (Optional)

Properties control much of the behavior of Content Server. You use the Property Editor to modify `futuretense.ini`. If the Property Editor does not appear (because you did not select it in the Installation Options window), after the installation completes, you can start the Property Editor manually and change values in the `.ini` file.

A complete description of all the properties in the `futuretense.ini` file is provided in the *CSEE Administrator's Guide*. Descriptions of each property are also available within the Property Editor itself. While the installation program is running, only a small fraction of the available properties are pertinent. The following table identifies the relevant properties:

Table 8: Properties to Consider Setting During Installation

Property	Setting
`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 some big number (for example, 100000).
`cc.datetime`	Set to `TIMESTAMP` if you are running Oracle 9. If you are running Oracle 8, accept the default value.
`cs.documentation`	The URL from which a user accesses Content Server documentation. See "Set Up Security" for details.

If your site uses JSP, then you must additionally consider the following properties. You can accept the default arguments, which are listed here:

Table 9: JSP Properties to Consider Setting During Installation

Property	Setting
`cs.jspclear`	Set to `true` to delete any previously deployed JSP files, and clear the WebSphere working folder (`.temp` and `.class` files) when the Content Server engine executes the first JSP deployed by Content Server. Otherwise, set to `false`.

`cs.jspwork`	If you set `cs.jspclear` to `false`, then Content Server ignores the value of `cs.jspwork`. If you set `cs.jspclear` to `true`, you must set `cs.jspwork` to the directory where WebSphere creates `.class` files. To determine this directory, examine the foll0
	owing file:
	`WebSphereRoot``/temp/`*`host_name`*`/cs/`
	`app_server_name``App.ear/jspwork`
	For example:
	`WebSphere/AppServer/temp/MyServer/cs/`
	`ContentServerApp.ear/jspwork`

`cs.jsproot`	This property holds the directory where WebSphere expects to find JSP source files. Set `cs.jsproot` to the following directory:
	`WebSphereRoot``/installedApps/`
	`app_server_name``App.earApp.ear/cs.war/jsp/`
	For example:
	`WebSphere/AppServer/installedApps/`
	`ContentServerApp.ear/cs.war/jsp/`

`cs.jsppath`	Set the `cs.jsppath` to the following value: `/jsp/` The `cs.jsppath` property is used in conjunction with `cs.jsproot`. To keep these properties in synch, you must set the `cs.jsppath` property to be the tail end of the value set for the `cs.jsproot` property, which is `/jsp/`.

Note

In some older versions of the Content Server Installer, you were instructed to modify the value of the `cc.security` property. As of Content Server 4.0, you should not alter the `cc.security` property during installation.

The resulting log file is in the Content Server installation directory; for example:

```
/local/ContentServer/futuretense.txt
```

When you finish making changes to the properties, choose **File** > **Save** and then **File** > **Exit**.

After the Property Editor window closes, the Content Server Installer resumes. Next, it displays the **iAS Install Actions** window.

Deploy the ContentServer.ear File

After starting the Content Server install and clicking through a series of screens, you will get a warning message indicating that you should restart the application server. When you get this message, do not continue the Content Server installation process until you finish deploying the `ContentServer.ear` file. After deploying the `ContentServer.ear` file, restart the application server and finish the Content Server installation.

1. Start the application server admin console.
2. Click **Applications**.
3. Click **Install New Application**.
4. In the **File Path** section, enter the path to the `ContentServer.ear` file (for example, `c:\FutureTense\ominstallinfo\app\ContentServer.ear`). Then, click **Next**.

> **Note**
>
> Do not use the Browse button to locate the `ContentServer.ear` file.

5. Click through the Bindings screen without configuring it by clicking **Next**.
6. On the **Enterprise Applications and Modules** screen, complete the following tasks:
   - Make sure that the **Distributed Application** box is checked.
   - If desired, select **Pre-Compile JSPs**.
   - Enter the application name, as specified in the **AppServer Root** dialog box in the Content Server installation.
   - Map the `cs.war` file to the appropriate virtual host, then click **Next**.
7. Map the module to the application server. Select **Server**, then check the box for `cs.war`.
8. Click **Apply**, then **Next**.
9. The **Summary** screen appears. Click **Finish**.
10. Start the web server using the following command:

    `startserver.sh` *`myservername`*

    where *`myservername`* is the name of your instance of WebSphere.

Restart WebSphere

To restart the WebSphere application server, do the following:

1. Run the `startupServer.sh` script in the */WebSphereInstallDir/*`bin` directory.
2. Verify that the server is running by viewing the `tracefile` in the */WebSphereInstallDir/*`logs` directory. The server is ready when the last line of the file appears as follows:

   ```
   server open for e-business
   ```

3. Run */WebSphereInstallDir*/bin/adminclient.sh to start the Advanced Administrative console.
   a. In the Advanced Administrative console. Expand the **Localhost** node.
   b. Right-click the Content Server application server.
   c. Select **Start** from the pop-up menu. A message box confirms that the server has been started.

Complete the Installation

Start the application server. After the application server has started, start the web server. Be sure to allow sufficient time for the application server to start (10-15 seconds) prior to starting the web server. If you proceed too quickly, you might encounter a "Could not bind to socket" condition.

Then, click **OK** on the **Install Options** window and the installation continues.

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

Finally, in the Installation Window itself, click **Exit** to finish.

Verify the Content Server Installation

To verify that the Content Server servlets were created successfully, do the following:

1. From a browser, enter the following URL:

   http://*localhost*/servlet/HelloCS

2. Your browser should display a success message.

Troubleshooting the Installation

To troubleshoot the Content Server installation, do the following:

1. Enable debugging in futuretense.ini, as described in "Change Properties (Optional)" on page 91.
2. Stop and restart the Content Server application server.
3. Rerun the installation procedure.
4. Examine the following files for errors:

   /WebSphereInstallDir/logs/contentserver_stderr.txt
   /WebSphereInstallDir/logs/contentserver_stdout.txt

   - If you see a message similar to "could not establish connection to jdbc driver," review "Creating the Data Source and Driver" on page 73.
   - If you see ORA-10536: space quota exceeded for tablespace x, check that the Oracle Content Server user has UNLIMITED TABLESPACE privileges.
   - If testing Content Server servlets returns a "document contains no data" error, review "Creating the Data Source and Driver" on page 73 to make sure you registered the servlets correctly. The contentserver_stdout.txt file should identify any servlets that failed to initialize.

- As a general rule with any error, ensure that your classpath is set correctly.

Optional Tasks After Installing Content Server

Perform the following tasks after you have installed Content Server:

- Set Up Security
- Enable Logging
- Overriding Default User Management for LDAP or Windows NT

Set Up Security

To set up security, perform the following tasks, which are detailed in the *CSEE Administrator's Guide*:

1. Add new users to the Content Server system.
2. Change the password for the Content Server system.
3. Secure web server document directories so that they allow only viewing of web documents (HTML, dynamic documents, or active server pages), not directories.
4. If necessary, change values of several security-related properties.

Enable Logging

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

The log file is in the `futuretense` subdirectory of the Content Server installation directory; for example, `export/home/FutureTense/futuretense`.

Note

Delete or archive this file often because a large log file can affect Content Server performance. Also note that enabling debugging can also affect performance. On a delivery system, you should disable debugging.

Overriding Default User Management for LDAP or Windows NT

For information about configuring your user management setup to use LDAP or Windows NT instead of the default facilities, see the *CSEE Administrator's Guide*.

Accessing Content Server Documentation

You can access Content Server documentation from two places:

- From a web site
- From the Content Server kit

Note

The most up-to-date documentation is on the web site, not on the kit.

Documentation on the Web Site

FatWire maintains a web site that contains the latest Content Server documentation, located at the following URL:

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

where *n* is the latest sub-version number; for example, to access the documentation for release 5.5.1, go to the following URL:

`http://e-docs.FatWire.com/CSEE/5.5.1/index.htm`

To ensure that you have the latest information, FatWire recommends that you use the documentation through this web site rather than through the documentation you received with your Content Server software. The web site also lets you easily download a package that contains all the latest documentation to your local site.

Documentation on the Kit

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

`DOC551 (in both .tar and .zip formats)`
: These files hold all the manuals associated with this release.

`ReadMe.htm`
: This HTML file contains the release notes.

The installation program does **not** install the documentation on your system. If you want to place this documentation on your system, then you must unpack it yourself. To unpack the `DOC551.zip` file, just use Winzip or an unzip utility. To unpack the `DOC551.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_55`:

```
$ tar -xvf DOC551.tar /local/CSEE_DOC_55
```

Setting Up Help

Most of the CSEE applications contain a help button (a large question mark symbol). When a user clicks the help button, Content Server redirects the user's browser to the URL stored in the `cs.documentation` property. By default, the `cs.documentation` property contains the URL of the FatWire documentation web site. If you prefer to get help from documentation stored locally, just change the value of the `cs.documentation` property to the local URL or to any URL at which Content Server documentation is stored.

Be sure that you check the Content Server web site periodically so you can download any new or revised documents to your local site.

Uninstalling Content Server

Uninstalling Content Server involves cleaning the following components:

- Content Server
- Application server
- DBMS

Content Server

To clean the disk on which Content Server was installed, do the following:

1. Delete the `omninstallinfo` directory, which is found in the HOME directory of the account that performed the Content Server installation.
2. Delete the entire Content Server installation directory and all its subdirectories.

Application Server

To clean the application server, do the following:

1. Start the application server administration console.
2. In the application server administration console, select **Nodes**.
3. Select the name of your node.
4. Select **Application Servers**.
5. Select the name of your application server. Remove it.
6. In the application server administration console, select **Resources**.
7. Select **JDBC Providers**.
8. Select the name of your JDBC driver. Remove it.

DBMS

The Content Server install creates many database tables. To uninstall Content Server completely, you should delete all the tables created by Content Server using your choice of database tools. Most sites create a separate DBMS account to hold Content Server information. So, the best way to locate all Content Server tables is to search for all tables owned by the DBMS account.

Chapter 11

Installing Content Server on a Cluster

This chapter provides an overview of installing *Content Server* in a cluster environment with IBM WebSphere Application Server. It contains the following sections:

- What Is a WebSphere Cluster?
- Overview of Cluster Installation Tasks
- Content Server Cluster Installation
- Post-Installation Tasks

This chapter does not include full instructions for installing a WebSphere cluster. Rather, it includes guidelines for the components to work in a cluster with Content Server. When you install the WebSphere cluster, use the vendor documentation as your primary source, and use the information in this chapter as a supplement.

What Is a WebSphere Cluster?

WebSphere Application Server is really an application server product that creates an environment in which application servers run, rather than an application server itself. In the WebSphere model, the Content Server product is imported as an enterprise application and assigned to run on an application server, which in turn runs under the WebSphere runtime. A single machine could have another application server that ran another application. Or it could have a clone of the first application server, which runs the same enterprise application that the first application server runs.

WebSphere Web Server Plug-in

The WebSphere web server plug-in forwards requests from a web server to a WebSphere runtime. Generally, you do not want to run an application server (which runs under a WebSphere runtime) on the same machine that you are using as a web server machine. If you have a WebSphere cluster, then you want your WebSphere Web Server Plug-ins to forward requests to all members of the cluster, not just to a single WebServer runtime on a single machine.

WebSphere Runtime

To run WebSphere Application Server, you run `/WebSphereInstallDir/bin/startupServer.sh` on a machine that has the WebSphere runtime installed. You should have only one copy of the WebSphere runtime installed on a machine. More than one application server can run under a single instance of WebSphere Runtime. For example, on a machine with four or more CPUs, you may want to run multiple clones of a single application server. You may even want to run a single machine with two or more different application servers that are not clones of a single server group.

While WebSphere runtime is running, it can handle requests that are forwarded to it from one or more WebSphere web server plug-ins.

WebSphere Cluster

When one or more WebSphere runtime instances all use the same database account for storage of their internal configuration data, they form a WebSphere cluster. Running WebSphere Administrative Console against any of the cluster members shows information about it, and allows control of all cluster members.

WebSphere Server Groups and Clones

WebSphere has a powerful concept of server groups and clones. A server group can be made from an application server, and that application server can be made into a clone of that server group. Then, additional clones can be made from that server group, all following the settings in the server group.

For example, server group 1 runs on Machine1. If a server group is made from that application server, then you can also run clones of server group 1 on Machine2 and Machine3. Now, three machines can all handle requests to server group 1. If Machine1 suffers a hardware failure, then Machine2 and Machine3 continue to handle the load, and users see very little, if any, interruption. The implications for ease of maintenance, configuration, and path for adding more powerful hardware, become obvious. Changing configuration information in the server group, allows an easy push of these configuration changes to the clones. Cluster members can be taken down, repaired, and replaced at will.

Typical clustered management and delivery systems are illustrated in the following diagram:

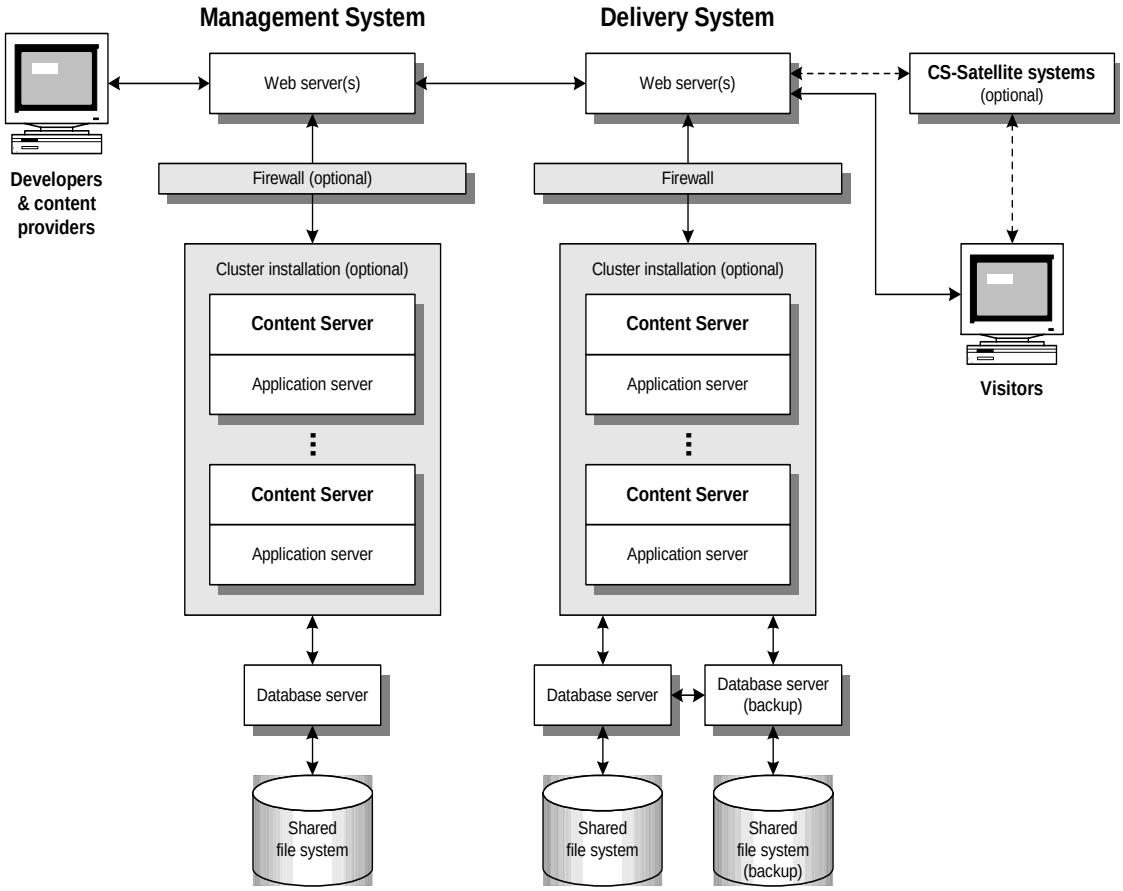


Figure 2: Typical clustered management and delivery site configurations

Overview of Cluster Installation Tasks

Carefully review the following top-level tasks for installing a Content Server cluster, and refer to the documentation supplied with your application server and operating system as needed for complete information.

1. Plan your WebSphere cluster installation.

   Review the WebSphere application server cluster installation guidelines before you actually install any software. See the IBM WebSphere InfoCenter for complete online documentation:

   `http://www-4.ibm.com/software/webservers/appserv/infocenter.html`

2. Create a shared file system (usually NFS) for use by Content Server.

   Follow instructions provided by your operating system vendor, and consider any applicable WebSphere application server guidelines.
3. Install the WebSphere application server (and IBM HTTP Server web server) on the cluster members according to the IBM WebSphere documentation.
4. Configure the WebSphere application server for Content Server as described in Chapter 8, "Configuring WebSphere for Content Server."
5. Install Content Server on the primary cluster member machine as described in Chapter 10, "Installing Content Server."
6. Create the WebSphere server group and clones.
7. Install Content Server on the other cluster member machine as described in Chapter 10, "Installing Content Server."
8. Set Content Server property values as described in "Additional Content Server Cluster Properties" on page 112.

Content Server Cluster Installation

This section contains instructions for installing Content Server in a WebSphere cluster environment, and lists several important tasks that you must complete before you actually install Content Server.

Pre-Installation Tasks

Before you install Content Server, perform the following tasks:

1. For Oracle, set up the database and Content Server database user as described in Chapter 3, "Installing Oracle."

   For DB2, set up the database and Content Server database user as described in Chapter 4, "Installing DB2."

   All cluster members use this Oracle database user.
2. Create a shared file system for sharing common files and synchronizing cache activities. For instructions, refer to your operating system documentation, and consider any applicable WebSphere application server guidelines.

   The shared file system usually (but not always) resides on the database machine, and on a fault-tolerant disk. If you replicate the database, you must replicate the shared file system as well. Make sure all machines in the cluster can read and write to the shared file system. The Content Server user account you create in step 6 of this procedure also must have the correct read/write privileges.

   Note

   Read/write privileges must be consistent across all cluster members.
3. Set up the WebSphere application server cluster according to instructions in the WebSphere documentation.

4. Set up the IBM HTTP web server according to the WebSphere documentation.
5. Synchronize system clocks for all machines involved in the cluster. Problems with synchronizing processes across cluster members are likely to occur if you do not do this correctly.

   Note

   FatWire recommends that you set up an automated or manual process that periodically synchronizes system clocks hourly or more frequently, depending on the accuracy of your system clocks.

6. Create a user account for installing Content Server. The user name, password, and privileges must be identical on each machine in the cluster. The Content Server installation creates files in the application server directory, so make sure this Content Server user has the correct privileges.

Installing Content Server on the Primary Cluster Member

To install Content Server on the primary cluster member:

1. Make sure that you have completed the Content Server pre-installation tasks described in “Pre-Installation Tasks” on page 102, and that you have correctly configured the application server and web server for Content Server.
2. Disable all cluster members except for the primary cluster member by stopping the application server on the other cluster members.
3. Log in to the primary cluster machine as the Content Server user you created in “Pre-Installation Tasks” on page 102.
4. Install Content Server according to the instructions in Chapter 10, “Installing Content Server,” except for the following cluster installation differences:
   a. In the **Installation Type** window, select **Single Server** as the type. This is the correct setting for a primary cluster member.
   b. In the **Web Server Document Root** window, enter the path where the shared folders are to be stored. This is the same shared directory you created in “Pre-Installation Tasks” on page 102.
   c. If the web server is on a different machine, specify the `futuretense_cs` web server document root on the shared file system. This directory is a temporary placeholder for Content Server web server files. You will copy these files later, as described in “Post-Installation Tasks” on page 111.

      During the Content Server installation on WebSphere 4.0.1, you imported the `ContentServer.ear` file into WebSphere as part of establishing the `ContentServerApp` Enterprise Application. Unlike the unclustered installation, however, you do not need to establish a new Enterprise Application to cluster Content Server across multiple machines. Instead, you turn the existing Content Server application server into a server group. Then, you configure WebSphere to clone that Server Group to another machine.
5. Start the WebSphere Administrative Console (if it is not already running).

6. From the WebSphere Administrative Console tree:
   a. Navigate to **Nodes > installNode > Application Servers > ContentServer**.
   b. Click **ContentServer**.
7. Stop the Content Server Application Server (if it is running):
   a. Right-click **ContentServer**.
   b. From the pop-up menu, choose **Stop**.
8. From the **Services** tab, choose **Session Manager Service**, and click **Edit Properties**. The **Session Manager Service** dialog box appears.
9. From the **Persistence** tab:
   a. Select the **Enable Persistent Sessions** check box.
   b. Select the **Medium (speed)** option.
10. From the **Database** tab:
    a. Choose the same data source that you set up for Persistent Session data:

       For DB2, the data source is separate from the data source on which you plan to store Content Server data.

       For Oracle, the data source may be the same as the data source on which you plan to store Content Server data.
    b. Specify the **Username**, **Password**, and **Confirm Password** that you want to use to store Persistent Session data in the specified Data Source.
    c. For **DB2 Row Size**, choose 4.
    d. In the **Table Space Name** field, enter `USERSPACE1`. For DB2, this name specifies default available tablespace for the DB2 database. For Oracle, the name is ignored.
    e. Select the **Use Multirow Sessions** check box.
11. Click **OK** to close the Session Manager Service dialog box.
12. If the Data Source was already listed in the **Database** tab, but the **Use Multirow Sessions** check box was not selected and you have enabled it, then you will have to drop the sessions table from that data source. You must drop the table from the database, not empty it. To do this, stop WebSphere and its application servers.

    As a test, start the Content Server application server to ensure that Content Server is still working. When you have determined that Content Server is working, stop it.
13. Back up your WebSphere configuration.

    You are about to make a significant change to the WebSphere configuration. At this time, FatWire recommends that you create a backup copy of WebSphere's current configuration, as follows:
    a. From WebSphere Administrative Console, choose **Console > Export to XML**.
    b. Choose a directory and file name that includes the `.xml` extension to store the current WebSphere XML configuration.

    If necessary, you can revert to your previous configuration by choosing **Console > Import from XML**. Note that reverting to the previous configuration works best when you have not made many changes after exporting to XML.

14. Create a server group.

    To enable multiple machines to handle requests to Content Server, you must complete these general tasks:

    - Turn the existing Content Server application server into a server group.
    - Make the original application server a clone of this server group.
    - Place clones on additional machines.

    To do this, complete the following steps:

    a. From the WebSphere Administrative Console, specify the tree view and navigate to **Nodes > installNode > Application Servers > ContentServer**.

    b. Right-click **ContentServer**, and choose **Create Server Group** from the pop-up menu. The **Server Group Properties** dialog box appears.

    c. From the **General** tab, enter `ContentServerGroup` in the **Server Group Name** field.

    d. From the **General** tab, click **Environment**, and verify that `/ContentServerInstallDir/bin` is specified as part of the `PATH/LIBPATH/LD_LIBRARY_PATH`. If this environment variable is not set, refer to "Creating the Content Server Application Server" on page 78.

    e. From the **General** tab, click **OK** to close the **Environment Editor** dialog box.

    f. From the **General** tab, choose **Module** in the **Module Visibility** field.

    g. From the **Advanced** tab, go to **Workload Management Selection** and choose **Round Robin Prefer Local as Policy**.

    h. From the **Services** tab, ensure that the Session Manager Service properties include the same persistence and database values that you set in the "Creating the Content Server Application Server" section.

    i. Click **OK** to create the server group.

    j. Click **OK** to dismiss the **Success** dialog box.

    k. Verify that the server group was created:

       In the WebSphere Administrative Console, verify that the **Server Groups** folder contains the **ContentServerGroup group**.

       Verify that your Application Server is listed as **ContentServer (ContentServerGroup)**. This name indicates that the Content Server application server is a clone of the new server group.

15. Regenerate the WebSphere web server plug-in configuration file. Note that each time you change the configuration of enterprise applications offered by an application server you must repeat this step. In this case, regenerating informs the web server that the newly created Server Group exists.

    a. In the WebSphere Administration Console tree view, navigate to **Nodes > *installNode***. (*installNode* is the name of the machine on which you want to install Content Server.)

    b. Right-click on your *installNode*, and choose **Regen Webserver Plugin** from the pop-up menu.

       Note that the WebSphere Administration Console may not inform you when the plug-in regeneration fails. If you suspect a problem occurred during regeneration,

you can get more information about it by regenerating from the command line. For example, enter the following UNIX command:

```
% /opt/WebSphere/AppServer/bin/GenPluginCfg.sh adminNodeName
installNode
```

Note that *installNode* is the application server machine on which `GenPluginCfg.sh`. runs.

16. If you run a single web server, and it is on the same machine as your application server, you can skip this step and continue. Otherwise, copy the new plug-in configuration file to each web server machine, as follows:
    a. Locate the `/WebSphereInstallDir/config/plugin-cfg.xml` file on your application server machine. (This is the same file that you generated in the previous step.)
    b. On each web server machine that serves pages for the application server machine, copy the `plugin-cfg.xml` file to the `/WebSphereInstallDir/config` directory.
17. If the web server is running, stop and restart it.
18. Test the Content Server Group and Content Server:
    a. Start the Content Server Group: Right-click `ContentServerGroup` in the tree, and select **Start** from the pop-up menu.
    b. To ensure that Content Server is still working, enter the following URL in a browser and verify the response:

       `http://localhost/servlet/HelloCS`
    c. Stop the Content Server Group: Right-click `ContentServerGroup` in the tree, and select **Stop** from the pop-up menu.
19. Share the Content Server shared directories.

    Remember that you installed the original Content Server application in three stages:

    - Imported the `/CSInstallDir/ominstallinfo/app/ContentServer.ear` file into WebSphere as an Enterprise Application that you named ContentServerApp.
    - Installed supporting files to disk, which we will refer to as location `/ContentServerInstallDir`, in this example. The `/ContentServerInstallDir` directory includes the following subdirectories: `/Content`, `/Storage`, `/elements`, and `/lock`.
    - Installed shared files to disk, which we will refer to as location `/ContentServerSharedDir`

    The `/ContentServerSharedDir` directory must be shared among the machines added to the Content Server cluster. For sharing, choose a file networking method, for example, NFS on Solaris or NFS on AIX. How you configure the sharing depends on whether you specified separate directories for the `/ContentServerInstallDir` and `/ContentServerSharedDir` during Content Server installation for the shared folder setting:

    - If you specified separate directories for `/ContentServerInstallDir` and `/ContentServerSharedDir` during the Content Server installation, then the operations that Content Server performs on the shared file system require a remote

file system. For this reason, do not place the file system locally on any application server installation. Because the application server and database are installed on different machines, FatWire recommends that you place the shared directory on the database machine. Keep in mind that if your database is not set up for automatic failover, then that machine is already a single point of failure. In this case, putting this shared directory on the database machine adds no additional point of failure. If the database has a failover machine, ensure that the file sharing product you choose enables you to perform file-system failover for the shared directory.

- If you did not specify separate directories for `/ContentServerInstallDir` and `/ContentServerSharedDir` during Content Server installation, then the Content Server installation directory must stay local to the application server machine. In this case, you cannot place the Content Server installation directory on a remote machine for sharing. Only the subdirectories of `/ContentServerSharedDir` should be shared across the cluster. The `/ContentServerSharedDir` directory includes the following subdirectories: `/Content`, `/Storage`, `/elements`, and `/lock`.

To share Content Server shared directories:

a. Create a directory on the database machine to hold the files from the shared directory.

b. Move the application server machine `/ContentServerSharedDir` directory to a backup location. For example, `/BackupDir`.

c. Mount the shared directory from the database machine to the application server machine on which you have already installed Content Server. Mount the directory in the same place that the `/ContentServerSharedDir` directory was prior to backing it up.

d. Copy all files in `/BackupDir` to the `/ContentServerSharedDir` directory that is newly mounted from the database machine. Although actually located on the database machine, the shared files are in the same location from the perspective of the application server machine.

20. Test Content Server and the Content Server group:

a. Start the Content Server group: Right-click `ContentServerGroup` in the tree, and select **Start** from the pop-up menu.

b. To ensure that Content Server is still working, enter the following URL in a browser and verify the response:

```
http://localhost/servlet/HelloCS
```

c. Stop the Content Server group: Right-click `ContentServerGroup` in the tree, and select **Stop** from the pop-up menu.

d. Delete `/BackupDir`.

21. Make this cluster member aware of other cluster members:

a. On the application server machine, start the Property Editor.

b. Open the `/ContentServerInstallDir/futuretense.ini` property file.

c. From the **Basic** tab, set `cs.session` to `true`. This enables session data to be in use.

d. From the **Cluster** tab, set the value of the `ft.sync`. synchronization ID to the same value across all members of this Content Server cluster. (Do not leave the value blank.) For example, the name of the database machine on which this cluster stores its data.

e. From the **Cluster** tab, set the value of the `ft.usedisksync` property to be the `/ContentServerSharedDir/lock` directory. If the `/lock` subdirectory does not exist in your `/ContentServerSharedDir` directory, create it now.

22. Test the Content Server group:

    a. Start the Content Server group: Right-click `ContentServerGroup` in the tree, and select **Start** from the pop-up menu.

    b. Stop the Content Server group: Right-click `ContentServerGroup` in the tree, and select **Stop** from the pop-up menu.

Installing Content Server on Other Cluster Members

To install Content Server on the other cluster members:

1. Make sure that you have completed the Content Server pre-installation tasks described in “Pre-Installation Tasks” on page 102.
2. Install WebSphere on the other cluster member machines. This is similar to installing WebSphere in the primary machine, but, consider the following:
   - Install only the WebSphere components that are required by the WebSphere installation process (Server Runtime, Administrative Console).
   - Use the same Oracle or DB2 database and user account that you used to install the WebSphere runtime on the primary cluster member. (All cluster members must store configuration information in the same database using the same account name.)
3. Mount `/ContentServerSharedDir` directory on the new cluster member machine, in the same place that you mounted it on the first cluster member machine. (Remember that this directory is shared from the database machine.)
4. Specify a database driver for the new cluster member machine:
   a. From the WebSphere Administrative Console tree, navigate to **Resources** > **JDBC Providers** > ***myDriver***, where ***myDriver*** is the driver name you chose in “Creating the Data Source and Driver” on page 73. Click ***myDriver***.
   b. From the **Nodes** tab, click on the node that is already listed. This is the first cluster member that you set up. The bottom of the **Nodes** tab should now display the classpath for the node that you chose. Remember this classpath—you will need to search for the same files on the new cluster member machine.
   c. Click **Install New**. The Install Driver dialog box appears.
   d. Click the node name of the new cluster member.
   e. Click **Specify Driver**. The Specify the Driver Files dialog box appears.
   f. From the Specify the Driver Files dialog box, click **Add Driver**, and find the first driver file. If you installed the driver file in the same place on this machine as on the first cluster member machine, then it is in the same location as the previous step.

g. Click **Open** when you have found the driver file. For DB2, you are only looking for one driver file. For Oracle, you are looking for two. Refer to the files that you previously noted.

h. Click **Add Driver** and add the next driver. Repeat this operation until all required driver files have been added to the list.

i. Click **Set** to close the Specify the Driver Files dialog box.

j. Click **Install** to close the Install Driver dialog box.

k. Click **Apply** below the **Nodes** tab to apply these configuration changes.

At this point, all data sources listed for this JDBC provider (including sources for storing persistent session data and Content Server data) are available on the new cluster member machine.

5. Use the WebSphere Administrative Console to create a new clone of the Content Server group:

   a. In the WebSphere Administrative Console tree view, navigate to **Server Groups > ContentServerGroup**. Right-click **ContentServerGroup**, and select **New > Clone**. The Create Clone dialog box appears.

   b. For **Node to Install Server On**, choose the new cluster member machine.

   c. In the **Name** field, enter `ContentServer`.

   d. Click **Create** to create the new clone.

   e. Click **OK** to dismiss the Success dialog box.

6. Log in to the secondary cluster machine as the Content Server user that you created in “Pre-Installation Tasks” on page 102.

7. Install Content Server according to the instructions in “Installing Content Server” on page 83, except for the following cluster installation differences:

   a. From the Installation Type window, select **Cluster Member** as the type.

   b. Unlike Content Server installation on the primary cluster member, you do not create a new Enterprise Application when you deploy the `ContentServer.ear` file on the cluster member machine. Instead, from the `/WebSphereInstallDir/installedApps` directory, enter the following command on a single command line:

      ```
      ../bin/EARExpander.sh
      -ear /CSInstallDir/ominstallinfo/app/ContentServer.ear
      -expandDir ./ContentServerApp.ear -operation expand
      -expansionFlags war
      ```

      On Windows NT, enter the following command:

      ```
      ..\bin\EARExpander.bat
      ```

      Note that if you are installing Content Server to the same directory on this machine as you did on the previous machine, then the `/CSInstallDir/ominstallinfo/app/ContentServer.ear` file on each machine will be the same. Alternatively, you could install the `ContentServer.ear` file from the original installation.

8. Regenerate the WebSphere web server plug-in configuration file, as follows. (Note that each time you change the configuration of enterprise applications offered by an application server you must repeat this step.)

   Note

   After you clone the Server Group, do not regenerate the web server plug-in configuration files until after you have expanded the `ContentServer.sh` file to the new cluster member. If you regenerate the web server plug-in configuration files between these two steps, then the file will contain no information about the Content Server group. Without this information, no requests can be forwarded to the Content Server group.

   a. In the WebSphere Administration Console tree view, navigate to **Nodes** > ***installNode***. The *installNode* is the machine on which you want to install Content Server.

   b. Right-click on your *installNode*, and choose **Regen Webserver Plugin** from the pop-up menu.

      Note that the WebSphere Administration Console may not inform you when the plug-in regeneration fails. If you suspect a problem occurred during regeneration, you can get more information about it by regenerating from the command line. For example, enter the following UNIX command:

      ```
      % /opt/WebSphere/AppServer/bin/GenPluginCfg.sh -
      adminNodeName installNode
      ```

      Note that *`installNode`* is the application server machine on which `GenPluginCfg.sh`. runs.

9. If you run a single web server, and it is on the same machine as your application server, skip this step and continue. Otherwise, copy the new plug-in configuration file to each web server machine, as follows:

   a. Locate the `/WebSphereInstallDir/config/plugin-cfg.xml` file on your application server machine. (This is the same file that you generated in the previous step.)

   b. On each web server machine that serves pages for the application server machine, copy the `plugin-cfg.xml` file to the `/WebSphereInstallDir/config` directory .

   c. Restart each web server so that it reads changes to its `/WebSphereInstallDir/config/plugin-cfg.xml` file.

10. Start the Content Server application server on the new cluster member machine.

11. Repeat the same steps that you completed for the primary cluster member. Pay particular attention to the following:

    a. In the Database Configuration window, enter the same database user and password you specified for the primary cluster machine.

    b. For the Content Server Shared Directory, choose the `/ContentServerSharedDir` directory that is now shared on the cluster member machine.

Post-Installation Tasks

Do the following after you complete the Content Server installation steps:

1. Stop all instances of WebSphere application servers.
2. On each cluster member, including the primary, start the Property Editor. (For more information about the Property Editor and property settings, see the *CSEE Administrator's Guide.)*

   a. From a system prompt, enter the following command on a single line or with line continuation characters:

   ```
   $ java -classpath cs-root/cs.jar:cs-root/swingall.jar
   COM.FutureTense.Apps.PropEditor /export/home/futuretense.ini
   ```

   b. Open the `futuretense.ini` property file (normally in the Content Server directory).
3. Set the following properties for each cluster member:

   `ft.sync` – To enable synchronization, set this to any value you want. This cannot be left blank, and must be identical on all machines in the cluster (for example, the DSN that cluster members use for their shared database).

   `ft.usedisksync` – Specify a shared file system folder for synchronizing data across a cluster. Set this to a valid empty folder (for example, a directory where the Content Server user has read/write access) when synchronization is enabled with the `ft.sync` property.

   `cc.security` – Set this to `true` to have Content Server check security before allowing database access. (Possible values are `true` and `false`.)

   These are the minimum Content Server properties required for clustering. Other properties that apply to clusters are described in "Additional Content Server Cluster Properties" on page 112.
4. Start all instances of WebSphere application servers.
5. Copy the `futuretense_cs` directory to all of the web server machines.
6. After you copy this directory, configure the Content Server web root directory as described in Chapter 8, "Configuring WebSphere for Content Server."

Note that if you are installing the CS Content Applications in addition to Content Server, there are additional post-cluster steps. For more information about installing the CS Content Applications on a cluster, see the *CS Content Applications Installation Guide*.

Testing the Cluster

To test the cluster, as installed so far:

1. Stop all instances of WebSphere application servers.

2. On each cluster member, including the primary member, start the Property Editor. (For more information about the Property Editor and property settings, see the *CSEE Administrator's Guide.)*
   a. From the system prompt, enter the following command on a single line or with line continuation characters:
      ```
      $ java -classpath cs-root/cs.jar:cs-root/swingall.jar
      COM.FutureTense.Apps.PropEditor /export/home/futuretense.ini
      ```
   b. Open the `futuretense.ini` property file (normally in the Content Server directory).
3. Set both the `ft.debug` and `ft.dbdebug` properties to `yes`.
4. Start all instances of the WebSphere application servers.
5. Create a simple test page. To do this, use Content Server Explorer to add a SiteCatalog entry that modifies a table row, then queries that table again.
6. On each cluster machine, tail the application server log file.
7. Open multiple browser windows, call the test page repeatedly in each window until you see all cluster members respond to the page request (you should see a database SQL statement in each log).

To test for failover:

1. Start a CS-Direct session and note which application server handles the requests.
2. Shut down that application server.
3. Verify that the session is maintained, and that a different application server picks up the requests.

Additional Content Server Cluster Properties

The Content Server `futuretense.ini` file contains properties that control various aspects of Content Server operations. You use the Property Editor to configure properties and set property values.

During installation, you typically use the Property Editor to verify and modify property settings needed by Content Server to identify the database and other components, such as the application server and debugger. After installation, you can use the Property Editor to performance-tune the application or to set specific values to enable clustering.

For each Content Server instance, make sure the following property values are set the same:

- Make sure all debugging is turned off for all Content Server instances. Start the Property Editor, and select the **Debug** tab on the left side. Make sure all of the debug properties are set to `no`.
- Select the **Basic tab**, and set `cs.timeout` to a desired value (in seconds).
- Select the **Caching** tab, and set `ft.filecheck` to `no`.
- Review all Caching properties and set the value accordingly. Good candidates to change include `cc.cacheResultsTimeout` and `cc.pgCacheTimeout`.

To improve performance or to stabilize write/read operations to the shared disk, the following property values can be set to a local folder:

- `cs.pgcachefolder` – Specifies the default locations for pages cached to disk. Located in the Content Server install folder.

  Note

  If you modify a cached page then it is possible that all of the cluster members may have different versions of the same page. If the cached page expires within a certain time, then the cache clears and the cluster members retrieve the new page from the database.

- `cs.pgexportfolder` – Specifies the default location for exported pages. Located in the Content Server `Install` folder.
- `cs.xmlfolder` – Specifies a working folder for use with HTML filtering. Located in the Content Server `Install` folder.

Appendix A

Pre-Installation Readiness

FatWire provides an Installation Services team to install Content Server. If you decide to hire this team, the manager of the team will contact you prior to performing the installation. During that contact, the manager will walk you through the pre-installation readiness checklist that appears in this appendix.

General Logistics

To help us plan, please provide the following information to us as soon as possible. In addition, please provide directions to your site and hotel recommendations.

Table 10: General Logistics

You Fill Out	Information We Need
	What is the address of the site where we will install the software?
	During which hours may the FatWire Installation Services representative perform the installation?
	Who is the primary contact at your organization? (Need name, e-mail address, and phone number.)
	Who is the secondary contact at your organization? (Need name, e-mail address, and phone number.)
	Have you received any training from FatWire? If so, what courses?
	What is your anticipated date to “go live”?
	Who is your system integrator?

Checklist: Before We Arrive

Before we arrive, you must have completed the tasks shown in the following table:

Table 11: Readiness Checklist

Completed?	Task
	All hosts have the operating system versions and patch levels required by Content Server.
	All hosts are configured with a static IP address and must be properly set up in DNS.
	You have sent the FatWire Installation Services team an architectural diagram. This diagram should identify the following: • The names and IP addresses of all hosts involved in the installation. • The software that will be installed on each host. • The position of any firewalls between hosts, or between the hosts and the rest of the network.
	A C compiler is installed on the host on which Apache will be installed. (If some other web server is to be installed, no C compiler is required.)
	If we are installing remotely (not on the console), then all of the following must be true: • A network drop to connect a laptop is available. The connection has been tested and works properly. (Be aware of firewall issues.) • Your firewall permits our laptop to access remote files via FTP. (We need FTP access in order to download software.)
	If we are installing on UNIX (Solaris or AIX), then we need at least one of the following: • The root password to all the hosts. • The name and phone number of the UNIX system administrator who can type in the root password when we need root access.
	An X-Windows client is installed on all UNIX hosts on which we will install Content Server. (Content Server is installed on the same host(s) running an application server.)
	If this is a multi-tiered installation with a firewall between the tiers, then the appropriate ports have been opened to allow communication between servers.
	If you are supplying the web server, you have the installation media and license keys.
	If you are supplying the application server, you have the installation media and license keys.
	If you are supplying the DBMS, you have the installation media and license keys.

Architecture Overview

We typically install software on three environments:

- Development
- Management
- Delivery

When we ask you to specify the number of tiers for each environment, your possible answers are as follows:

- One tier – all software is installed on a single host.
- Two tier – software is installed across two hosts. Typically, the web server and application server are installed on one host, and the DBMS is installed on a different host.
- Three tier – software is installed across three hosts. The web server, application server, and DBMS are all installed on different hosts.

We also need to know if any of the tiers are clustered on each environment. Clustering means that the same component is installed on multiple hosts. For example, application servers are often clustered, meaning that the application server is installed on two or more hosts.

Development Environment

Please complete Table 12, Table 13, and Table 14 for your development environment.

Table 12: Architecture for Development Environment

You Fill In	Parameter
	How many tiers? (Typically, there is only one.)
	Is there any clustering? (Typically, there is not.) If there is clustering, describe which components are clustered.
	What is the operating system version, including patch level or service pack.

Table 13: Hardware for Development Environment

Hostname	Purpose (for example, DBMS host)	Make/Model	Processor Speed	RAM

Table 14: IP Addresses for Development Environment

Hostname	IP Address (primary/front-end)	IP Address (secondary/back-end)

Management Environment

Please complete Table 15, Table 16, and Table 17 for your management environment.

Table 15: Architecture for Management Environment

You Fill In	Parameter
	How many tiers?
	Is there any clustering? If there is clustering, describe which components are clustered.
	What is the operating system version, including patch level or service pack.

Table 16: Hardware for Management Environment

Hostname	Purpose (for example, DBMS host)	Make/Model	Processor Speed	RAM

Table 17: IP Addresses for Management Environment

Hostname	IP Address (primary/front-end)	IP Address (secondary/back-end)

Delivery Environment

Please complete Table 18, Table 19, and Table 20 for your delivery environment.

Table 18: Architecture for Delivery Environment

You Fill In	Parameter
	How many tiers?
	Is there any clustering? If there is clustering, describe which components are clustered.
	What is the operating system version, including patch level or service pack.

Table 19: Hardware for Delivery Environment

Hostname	Purpose (for example, DBMS host)	Make/Model	Processor Speed	RAM

Table 20: IP Addresses for Delivery Environment

Hostname	IP Address (primary/front-end)	IP Address (secondary/back-end)

Software

A full installation requires the following components:

- Web Server
- Application Server
- DBMS
- JDBC Driver
- FatWire Products

Some of these components might already be installed on your systems. We need to know what is already installed and what needs to be installed.

Web Server

Is a web server already installed? If so, complete Table 21. If you want us to install a web server, complete Table 22. (You might need to complete both tables.)

Table 21: Web Server Already Installed

You Fill In	Parameter
	Type of web server (for example, Apache)
	Version number (for example, 1.3.12)
	Has connectivity to the application server (plug-in/web connector) been configured and tested?
	Is there a single web server or are there multiple web servers with load balancing?

Table 22: Web Server That We Will Install

You Fill In	Parameter
	Type of web server (for example, Apache)
	Version number of web server
	If we upgrade the web server described in Table 21, do you want us to preserve the current configuration settings (as much as possible)?

Application Server

Is an application server already installed? If so, complete Table 23. If you want us to install an application server, complete Table 24. (You might need to complete both tables.)

Table 23: Application Server Already Installed

You Fill In	Parameter
	Type of application server (for example, WebSphere)
	Version number (for example, 5.1.0) Be precise about patch numbers or service packs.
	What applications is this application server already running?
	Are there currently any other applications on this host that will share hardware resources?
	How many hosts are running application servers in this environment?

Table 24: Application Server That We Will Install

You Fill In	Parameter
	Type of application server (for example, WebSphere)
	Version number of application server.
	If we upgrade the application server described in Table 21, do you want us to preserve the current configuration settings (as much as possible)?

DBMS

Is a DBMS already installed? If so, complete Table 25. If you want us to install a DBMS, complete Table 26. (You might need to complete both tables.)

Table 25: DBMS Already Installed

You Fill In	Parameter
	Type of DBMS (for example, Oracle)
	Version number of DBMS (for example, 8.1.7) Be precise about patch numbers or service packs.
	Have tablespaces and user IDs been created for Content Server?
	Is this DBMS clustered? If so, on how many hosts in this environment?

Table 26: DBMS We Will Install

You Fill In	Parameter
	Type of DBMS (for example, Oracle)
	Version number. Be precise about patch numbers or service packs.
	Will we have the DBMS system password? If not, who is the database administrator to contact during the installation?
	Will this DBMS need to store non-English content?
	Is this DBMS shared used by any other applications?

JDBC Driver

Is a JDBC driver already installed? If so, complete Table 27. If you want us to install a DBMS, complete Table 28. (You might need to complete both tables.)

Table 27: JDBC Driver Already Installed

You Fill In	Parameter
	Type of JDBC driver currently installed (for example, Oracle Type 2)

Table 28: JDBC Driver We Will Install

You Fill In	Parameter
	Type of JDBC driver we should install (for example, Oracle Type 4)
	Does the DBMS store non-English content?

FatWire Products

Please complete the following table, identifying those products currently installed and those you want us to install.

Table 29: FatWire Products

Currently Installed Version	Version That We Should Install	Product Name (Former Product Names are in Parentheses)
		Content Server
		CS-Direct (Content Centre)
		CS-Direct Advantage (Catalog Centre)
		CS-Engage (Marketing Studio)
		CS-Satellite (Satellite Server)
		CS-Bridge Enterprise (Integration Centre)
		Analysis Connector
		Commerce Connector
		A search engine, such as AltaVista

Index

A

administrative console 100

B

books 96

C

cc.datetime property 91
character sets
  configuring for DB2 39
  configuring for Oracle 26
  configuring for SQL Server 58
  setting JVM options 79
checking
  Oracle installation 60
cloning a model 100
cluster 87
  cloning a model 100
  Content Server installation 103
  members 108
  pre-installation tasks 102
  primary member 103
  properties 112
  testing 111, 112
  top-level tasks 101
configuring
  character sets 14
  LDAP 14
Content Server
  cluster installation 103, 108
  configurations 13
  creating a tablespace 27
  documentation 96
  installation file (AIX) 84
  installation file (Solaris) 85
  installation file (Windows NT) 86
  installation log file 95
  Oracle user account 30
  post-installation tasks 111
  primary cluster member 103
  Property Editor 112
  troubleshooting 94
  web root directory 103
Content Server database
  DB2 46
  Oracle 28
ContentServer.ear file 93

D

Database Users screen 90
databases
  creating DB2 Content Server 46
  creating DB2 persistent sessions 51
  creating Oracle Content Server 28
  creating Oracle persistent sessions 29
  installing Oracle 25, 57
  testing installation 34
DB2
  installing with WebSphere 40
debugging
  relevant properties 91

servlet 88
Deploy Debug Servlet 88
deploy debug servlet 88
documentation 96

E

ear file, installing 93

F

futuretense.ini properties 91

H

hardware
requirements 12
help 96

I

installation
planning 17, 25, 57
work sheets, work sheets, installation 17
installation log file 95
installing
clusters 87
documentation 96
JDBC driver for SQL Server 61
SQL Server 2000 58
installing Content Server
Application Server screen 88
clusters 87
Content Server Configuration screen 89
ContentServer.ear file 93
Database Configuration screen 90
Database Users screen 90
Installation Directory screen 87
Installation Options screen 88
Installation Type screen 87
Warning screen 90
Web Server Configuration screen 88
Web Server Document Root screen 89
WebSphere Root Directory screen 90
installing Portal Example 88
internationalization
configuring character sets (DB2) 39
configuring character sets (Oracle) 26
configuring character sets (SQL Server) 58
setting JVM options for character sets 79

J

JVM options
setting character sets 79

L

language support 14
latest documentation 96
LDAP user management 14

M

manuals, online 96

O

online help 96
online manuals 96
Oracle
configuration recommendations 26
creating tablespace 27
creating user account 30
installing on Solaris 25, 37, 57
installing on Windows 25, 57
installing with Enterprise Manager 28
testing database installation 34, 60
verifying installation 60

P

persistent sessions database
DB2 51
Oracle 29
Portal Example 88
portal example 88
primary cluster member 103
properties
futuretense.ini 91
Property Editor 88, 112
starting 111, 112

R

requirements
hardware 12

S

Select Products screen 87
starting

Property Editor 111, 112

T

tablespace
creating Content Server temporary 30
creating default for Content Server 28
creating default for persistent sessions 28
creating persistent session temporary 30
creating using SQL 29
creating WebSphere temporary 30
testing
cluster 111, 112
troubleshooting
Content Server 94

U

utilities
Oracle Enterprise Manager 28
Oracle SQL*Plus 29

V

validating
Oracle installation 60

W

web root directory 103
web server plug-in 99
WebSphere
administrative console 100
configuring for Content Server 69
creating application servers 78
installation prerequisites 65
runtime 100
web server plug-in 99
WebSphere (Solaris/AIX)
configuring Oracle8i for Content Server and WebSphere 36
restarting 93
testing servlets 94
WebSphere (Windows)
restarting 93
testing servlets 94