FatWire | Content Server 7

Version 7.0.2

Installing Content Server with JBoss Application Server and JBoss Portal

Document Revision Date: Jun. 15, 2011

FatWire SOFTWARE

Installing Content Server with JBoss Application Server and JBoss Portal
Document Revision Date: Jun. 15, 2011
Product Version: 7.0.2

FatWire Technical Support
www.fatwire.com/Support

FatWire Headquarters
FatWire Corporation
330 Old Country Road
Suite 303
Mineola, NY 11501
www.fatwire.com

Table of

Contents

Part 4. Content Server

Chapter 1

Introduction

This document provides guidelines for installing Content Server on JBoss Application Server 4.2.0 and JBoss Portal 2.6.1, connecting to a supported database of your choice.

Note

Anyone using this guide is expected to have experience installing and configuring databases, web servers, and application servers. Selected information regarding the configuration of third-party products is given in this guide. For detailed information about a particular third-party product, refer to that product's documentation.

This chapter provides information that will help you prepare for the Content Server installation. It contains the following sections:

About This Guide

This guide covers the installation, configuration, and maintenance of JBoss Application Server and JBoss Portal, as required to support Content Server. This includes the configuration of one or more JBoss instances, clustering, SSL, and backend databases.

How This Guide Is Organized

The content of this guide is organized by function rather than the order in which installation steps are completed. For example, a function such as application deployment is associated with the application server. It is presented in Part II (which covers the application server), even though it is performed later, when Content Server is installed (Part IV). Each major component of the Content Server installation is covered in its own part. A summary of the installation steps in the required order is given at the end of this chapter (see the section, ““Installation Quick Reference,” on page 7).

Terms and Acronyms

`<jboss_install_dir>`	Path to the JBoss installation directory.
`<cs_install_dir>`	Path to the Content Server installation directory.

Graphics in This Guide

Many steps in this guide include screen captures of dialog boxes and similar windows that you interact with in order to complete the steps. These screen captures are presented to help you follow the installation process. They are not intended to be sources of specific information, such as parameter values, options to select, or product version number.

Installation Quick Reference

After you install and configure the J2EE components that support Content Server, you will run the Content Server installer, which will guide you through the installation process. You will run the installer on each development, delivery, and management system on which you plan to use Content Server. During the Content Server installation, you will have the option to install sample sites and sample content.

Note

The names of the systems in your Content Server environment might differ from the names used in this document. Typically, the management system is also called "staging," and the delivery system is also called "production."

The following steps summarize the installation and configuration of Content Server and its supporting software. Keep the steps handy as a quick reference to the installation procedure and to the chapters that provide more detailed instructions.

I. Set Up the Database

Install, create, and configure your choice of supported databases. For instructions on creating and configuring the database, see *Configuring Third-Party Software*.

II. Set Up the Application Server

Set up JBoss Application Server, as shown in Chapter 3, "Installing JBoss Application Server and JBoss Portal." The steps you will complete are the following:

1. Installing JBoss (page 18).
2. Setting Up the Environment (page 20). This means creating a directory for Content Server, creating a JBoss instance, and testing the JBoss instance.

   Notes

   If you plan to install a vertical cluster, you will repeat this step to create a JBoss instance for each cluster member. Before creating multiple JBoss instances, however, it is best to first create one instance (as done in this guide), install Content Server, and verify that it is functional.

   If you are installing the JBoss portal, you must install and configure a supported LDAP server (OpenLDAP in this guide). You can do so in the current step, or after the first instance of Content Server is installed (as done in this guide). Instructions for installing and configuring LDAP servers are given in *Configuring Third-Party Software*.

3. If you are already running one or more instances of JBoss on your system (regardless of the application they are serving), or if you are planning to set up a Content Server cluster, ensure that each JBoss instance runs on a unique port number.
   For instructions, see "Setting Up Multiple Instances of JBoss to Run Simultaneously," on page 24.

III. Set Up the Web Server

This step is required if:

- You wish to use Apache web server
- You are creating a vertical cluster that will be load balanced through Apache web server
- You are planning to configure JBoss to run over SSL through Apache Web Server

To set up the web server, complete the following steps (in Chapter 4):

1. Integrating JBoss with Apache 2.0.x (page 28).
2. For SSL support, also complete the steps in "Configuring JBoss for SSL through Apache 2.0.x," on page 29.

> **Note**
>
> For single-server installations, installing a web server is optional. You can install the web server at any time prior to or following the Content Server installation.

IV. Install and Configure Content Server

In this step, you will first make sure that the listed pre-requisites for installing Content Server are satisfied. You will then install Content Server, verify the installation, and complete any additional post-installation steps that apply to your system (for example, set up a vertical cluster or configure the JBoss portal).

1. Before installing Content Server, do the following:
   a. Start the JBoss instance.
   b. Create a valid directory into which you will install Content Server. A valid directory is one that does not contain spaces in its name and has write permissions.
   c. For clustered installations, create a shared file system directory that all cluster members can read from and write to; the directory name and path cannot contain spaces. Note the following:
      - For delivery systems, the default location of the shared file system directory is the directory containing the directory in which Content Server is installed.
      - For content management and development systems, the default location of the shared file system directory is inside the directory in which Content Server is installed.
2. Install and deploy Content Server by running the supplied installer. The installer provides online help at each screen, should you need guidance. For more information, see "Installing Content Server," on page 36.

   If you are using an Oracle database and will require text attributes greater than 2000 characters, you will have to the `cc.bigtext` property to `CLOB` after the CS application is deployed half-way through the installation. For instructions, see step - in "Running the Installer," on page 36.

3. Complete the Content Server installation by performing the following steps:
   a. If you installed Content Server on Unix, set the permissions for Content Server binaries by following the steps in "Setting File Permissions (Unix Only)," on page 37.
   b. Verify the Content Server installation by logging in as the administrator. For instructions, see "Verifying the Installation," on page 37.
   c. If you need to perform LDAP integration, follow the steps in "Integrating with LDAP (Required for Portal Installations)," on page 43. LDAP integration is mandatory for portal installations, and optional for web installations.
   d. If you are creating a portal installation, set up the JBoss portal by following the steps in "Configuring JBoss Portal for Content Server (Required for Portal Installations)," on page 44.
   e. If you are creating a vertically clustered system, follow instructions in "Setting Up a Content Server Cluster (Optional)," on page 57.
   f. Once the entire installation is completed and verified, set up Content Server for its business purpose. For instructions, see the *Content Server Administrator's Guide* and the *Content Server Developer's Guide*.

Part 1

Database

This part contains a short chapter summarizing the databases that Content Server uses. Instructions on creating and configuring the databases are given in *Configuring Third-Party Software.*

This part contains the following chapter:

- Chapter 2, "Setting Up a Database"

Chapter 2

Setting Up a Database

Content Server requires access to a database that is specifically configured for Content Server. The list of supported databases (as well as other third-party components) is given in the *Supported Platform Document*, accessible from:

`http://e-docs.fatwire.com/CS`

(Click the Content Server version number, and on the Content Server page, click the **Supported Platform Document** link.)

Before installing any other of Content Server's supporting software, you must complete the following steps:

1. Install the database management system.
   For instructions, refer to the product vendor's documentation.
2. Create and configure a database for Content Server.
   For instructions, consult our guide *Configuring Third-Party Software.* Note that database configuration is identical across different application servers. Refer to the correct chapter to create and configure the database of your choice.

Part 2

Application Server

This part contains information about installing and configuring JBoss Application Server. This part contains the following chapter:

- Chapter 3, “Installing JBoss Application Server and JBoss Portal”

Chapter 3

Installing JBoss Application Server and JBoss Portal

This chapter provides instructions for installing JBoss Application Server and JBoss Portal so far as needed to install and run Content Server as a single instance or in a vertical cluster. For more information about the JBoss installation process, see the documentation that comes with JBoss.

This chapter contains the following sections:

Start/Stop Commands

Note

All commands are based on the assumption that `JAVA_HOME` is set to the proper directory.

- To start the JBoss instance:
  - For JBoss 4.0.5:

    ```
    <jboss_install_dir>/bin/run.sh -c <instance_name>
    ```
  - For JBoss 4.2.0 and JBoss Portal:

    ```
    <jboss_install_dir>/bin/run.sh -c <instance_name>
       -b <hostname>
    ```
- To shut down the JBoss instance:

  ```
  <jboss_install_dir>/bin/shutdown.sh -s jnp://<hostname>:
     <jnp port>:
  ```

Table 1: Port Configurations and Numbers

Port Configuration	Web Service Port	JNP Port
ports-default	8080	1099
ports-01	8180	1199
ports-02	8280	1299
ports-03	8380	1399

Step I. Installing JBoss

To install JBoss Application Server and JBoss Portal

1. Create a JBoss installation directory.
2. Decompress JBoss into the JBoss installation directory. Follow the examples below, but replace the file names with names that correspond to your version of JBoss:
   - For JBoss Application Server, decompress the `jboss-4.2.0GA.tar.gz` file into the JBoss installation directory as follows:
     1) `gzip -d jboss-4.2.0GA.tar.gz`
     2) `tar -xvf jboss-4.2.0GA.tar`
   - For JBoss Portal, decompress the `jboss-4.2.0GA.tar.gz` and `jboss-portal-2.6.1.zip` files into the JBoss installation directory as follows:
     1) `gzip -d jboss-4.2.0GA.tar.gz`
     2) `tar -xvf jboss-4.2.0GA.tar`
     3) `jar -xvf jboss-portal-2.6.1.zip`

3. If you are creating a portal installation, follow the instructions below (otherwise, skip to step 4):
   a. Move the `jboss-portal.sar` directory from `jboss-portal-2.6.1` to the `jboss-4.2.0.GA/server/default/deploy` directory.
   b. Copy the `Portal-`*`dbType`*`-ds.xml` file from `jboss-portal-2.6.1/setup` to `jboss-4.2.0.GA/server/default/deploy` (where *`dbType`* is the type of database you are using).
4. Set `JAVA_HOME` to the `jdk` folder of the version of Java that will be used. The Java version must be 1.5 or higher.

   `Ex:` **`export JAVA_HOME=/opt/jdk1.5.0_06`**
5. If you are creating a portal installation, create a database that JBoss Portal will use to store its data. For instructions on creating a database, see our guide, *Configuring Third-Party Software*. When creating the database, follow these steps:
   a. Use the same permissions as for the Content Server database.
   b. When you have created the database, edit the `Portal-`*`dbType`*`-ds.xml` file to point to it. For instructions, see step 4 on page 22.
6. Start the application server:

   `<jboss_install_dir>/bin/run.sh -b <hostname>`

   This will start the default JBoss instance on port 8080.
7. Test your JBoss installation:
   a. Point your browser to the following URL to test whether JBoss Application Server is running:

      `http://<hostname>:8080/`
   b. When the default "JBoss Application Server" home page is displayed, click **JBoss Web Console** to display the JBoss Management Console.
   c. If you are creating a portal installation, point your browser to the following URL to test whether JBoss Portal is running:

      `http://<hostname>:8080/portal`
8. When JBoss is successfully installed, continue with "Step II. Setting Up the Environment," on page 20.

Step II. Setting Up the Environment

In this section, you will create a directory for Content Server, create a new JBoss instance, and test the new instance.

Note

If you are installing a cluster, complete the steps below for each cluster member. The cluster must be vertical.

1. Create a directory for Content Server (for example `CS702`):

   ```
   mkdir CS702
   ```

2. Create a new JBoss instance:

   ```
   cd <jboss_install_dir>/server/
   ```

   As this instance has all the properties of a JBoss **all** instance, copy the **all** instance folder:

   ```
   cp –R all fatwire
   ```

   In this guide, we will call the new instance `fatwire`. However, you may name the instance as you wish.

3. In Table 2, on page 21, find the data source information for the database you will be using, and place the required files in:

   `<jboss_install_dir>/server/<instance name>/lib`

Table 2: Data Source Information

Database Driver	Parameter	Value
JTDS (third-party driver)	DriverClass	`net.sourceforge.jtds.jdbcx.JtdsDataSource`
	Required `.jar` files	`jtds-1.2.jar`
	URL	`jdbc:jtds:sqlserver://<server>:<dbport>/<dbname>` **Ex:** `jdbc:jtds:sqlserver://10.120.14.22:1433/CS70`
DB2	DriverClass	`com.ibm.db2.jcc.DB2Driver`
	Required `.jar` files	`db2jcc.jar, db2cc_license_cu.jar`
	URL	`jdbc:db2://<hostname>:<dbport>/<dbname>` **Ex:** `jdbc:db2://10.120.16.30:50001/CS70`
Oracle	DriverClass	`oracle.jdbc.driver.OracleDriver`
	Required `.jar` files	`ojdbc14.jar`
	URL	`jdbc:oracle:thin:@//<hostname>:1521/<dbname>` **Ex:** `jdbc:oracle:thin:@//godzilla.fatwire.com:1521/CS70`

Note

If you are using an Oracle database and require text attributes greater than 2000 characters, you will have to set `cc.bigtext` to `CLOB`. To support `CLOB`, use Oracle database 9.2.0.6 (or a higher supported version). Also use Oracle 10g drivers. (`CLOB` is not supported for lower database versions and for Oracle drivers 9x [thin, type 4].)

You will set `cc.bigtext` to `CLOB` when you run the Content Server installer (as explained in "Running the Installer," on page 36.)

4. Create a new data source file:
   a. Create a new data source file named `<database_type>-ds.xml` in `<jboss_install_dir>/server/<instance_name>/deploy`
   b. Paste in the XML code shown below:

```
<?xml version="1.0" encoding="UTF-8"?>
   <datasources>
     <local-tx-datasource>
       <jndi-name>jdbc/csDataSource</jndi-name>
       <connection-url><URL> </connection-url>
       <driver-class><Driver Class></driver-class>
       <user-name><User></user-name>
       <password><Password></password>
     </local-tx-datasource>
   </datasources>
```

   c. Replace the **<URL>** and the **<Driver Class>** text fragments with the URL and driver class selected from Table 2, on page 21.
   d. Replace the **<User>** and **<Password>** text fragments with the user information that you connect to your database with.
5. Do one the following, depending on your configuration:

- **For JBoss 4.0.5:**
  1) Open the file `<server_name>\deploy\jbossweb-tomcat55.sar\conf\web.xml` in a text editor.
  2) Uncomment the following section (that is, delete the two comment lines shown in bold type below):

```
<!-- Uncomment to use jdk1.5 features in jsp pages
   <init-param>
      <param-name>compilerSourceVM</param-name>
      <param-value>1.5</param-value>
   </init-param>
-->
```

- **For JBoss 4.2.0 and JBoss Portal:**
  1) Open the file `<server_name>\deploy\jboss-web.deployer\conf\web.xml` in a text editor.
  2) Add the code in bold to the JSP page compiler and execution servlet parameters:

```
<servlet>
   <servlet-name>jsp</servlet-name>
   <servlet-class>
     org.apache.jasper.servlet.JspServlet
   </servlet-class>
   <init-param>
      <param-name>compilerSourceVM</param-name>
      <param-value>1.5</param-value>
   </init-param>
   <init-param>
      <param-name>compilerTargetVM</param-name>
      <param-value>1.5</param-value>
   </init-param>
```

3) Add the code in bold to the common context parameters section:

```
<!-- JBossInjectionProvider provides resource
 injection for managed beans. -->
<!-- See JSF 1.2 spec section 5.4 for details. -->
<context-param>
   <param-name>
     org.jboss.jbossfaces.WAR_BUNDLES_JSF_IMPL
   </param-name>
   <param-value>true</param-value>
</context-param>
```

6. **(JBoss 4.2.0 only)** Edit the `run.sh` script located in `<jboss_install_dir>/bin` by adding the following line at the beginning of the file (after the comments section):

   ```
   JAVA_OPTS="-
      Dorg.apache.catalina.STRICT_SERVLET_COMPLIANCE=false
      $JAVA_OPTS"
   ```

7. Test the new instance by starting it and connecting to the "JBoss Management Console." Do the following:

   a. Ensure that the default instance is shut down before trying to start the new one. The instance needs to be shut down if you can view `http://<hostname>/web-console/` in your browser. Use the following command to shut down the default instance:

      `<jboss_install_dir>/bin/shutdown.sh -s jnp://<hostname>:1099`

   b. Start the new instance with the following command:

      `<jboss_install_dir>/bin/run.sh -c fatwire -b <hostname>`

   c. Access the "JBoss Management Console" using a web browser to confirm that JBoss is running:

      `http://<hostname>:8080/web-console/`

8. (Optional) If you wish to run multiple instances of JBoss simultaneously, continue with "Step III. Setting Up Multiple Instances of JBoss to Run Simultaneously," on page 24.

Step III. Setting Up Multiple Instances of JBoss to Run Simultaneously

Note

If multiple JBoss instances will be run simultaneously, you must ensure that no two instances are using the same port. Complete the steps below for all JBoss instances that must run simultaneously—**regardless of the applications they serve**.

1. Shut down all JBoss instances:

   <jboss_install_dir>/bin/shutdown.sh -s jnp://<hostname>:1099

2. No two JBoss instances can use the same ports. Change the ports as follows:

   a. Change to the `/conf` directory of your instance:

      cd <jboss_install_dir>/server/<instance_name>/conf

   b. Open the `jboss-service.xml` file in a text editor and find the "Service Binding" section. Look for the following `XML` code:

```
<mbean
  code="org.jboss.services.binding.ServiceBindingManager"
   name="jboss.system:service=ServiceBindingManager">
  <attribute name="ServerName">ports-01</attribute>
  <attribute name="StoreURL">${jboss.home.url}/docs/
    examples/binding-manager/sample-bindings.xml</attribute>
   <attribute name="StoreFactoryClassName">
    org.jboss.services.binding.XMLServicesStoreFactory
   </attribute>
</mbean>
```

      The `StoreURL` attribute holds the path to a file, which by default contains four different port configurations. These configurations are named `ports-default`, `ports-01`, `ports-02`, and `ports-03`.

      - To access these configurations, uncomment the code above (by deleting the "close comment" tag (`-->`) at the end of the code and inserting it above the opening `<mbean` tag).
      - The `ServerName` attribute holds the value of the port configuration that this instance will be using. To use a configuration other than `ports-01`, change the value according to the table in Table 1, on page 18.
      - To view which ports each configuration uses, view the `<jboss_install_dir>/docs/examples/binding-manager/sample-binding.xml` file.

3. If the instances are not already running, start them with the command:

   <jboss_install_dir>/bin/run.sh -c <instance_name> -b <hostname>

4. Test that two instances can be run simultaneously by connecting your browser to `http://<hostname>:<new_instance_port>/web-console/` to display the "JBoss Management Console." If the `ports-01` configuration is being used, the port will be `8180`.

Part 3

Web Server

This part shows you how to integrate Apache Web Server with JBoss, a requirement if you are creating a clustered installation or using SSL.

This part contains the following chapter:

- Chapter 4, “Installing and Configuring Apache Web Server”

Chapter 4

Installing and Configuring Apache Web Server

This chapter contains information about integrating JBoss with Apache 2.0.x and configuring JBoss to run over SSL through Apache Web Server. It contains the following sections:

Note

The steps in this chapter are required only if you are creating a clustered installation or you are planning to use SSL through Apache Web Server.

Integrating JBoss with Apache 2.0.x

This section shows you how to configure Apache 2.0.x to integrate with JBoss.

Note

Complete this section if you are setting up a Content Server cluster or using SSL through Apache Web Server.

- For clustered installations, this section is a pre-requisite for the load balancing procedure you will perform ("Balancing Load with mod_jk," on page 58).
- For SSL, this section is also a pre-requisite for the next section, "Configuring JBoss for SSL through Apache 2.0.x," on page 29.

1. Install Apache 2.0.x.
2. Set `$APACHE2_HOME` to the directory in which Apache was installed.
3. Download and compile the newest release of `mod_jk`.
   a. After downloading `mod_jk`, untar it using the commands:

      ```
      gunzip jakarta-tomcat-connectors-1.2.14.1-src.tar.gz
      tar -xvf jakarta-tomcat-connectors-1.2.14.1-src.tar
      ```

   b. Go to the directory `jakarta-tomcat-connectors-<version>-src/jk/native`:

      ```
      cd jakarata-tomcat-connectors-1.2.14.1-src/jk/native
      ```

   c. Configure and compile the `mod_jk.so` file:

      ```
      ./configure --with-apxs=$APACHE2_HOME/bin/apxs
      make
      cd apache-2.0
      $APACHE2_HOME/bin/apxs -n jk -i mod_jk.so
      ```

      The last command will automatically place the `mod_jk`.so file into your `$APACHE2_HOME/modules` directory.
4. Create `workers.properties` in `$APACHE2_HOME/conf` with the following contents:

   ```
   ps=/
   worker.list=jboss

   worker.jboss.port=<ajp port>
   worker.jboss.host=<hostname>
   worker.jboss.type=ajp13
   worker.jboss.lbfactor=1
   ```

   (The ajp port can be found in `<jboss_install_dir>/server/<instance name>/deploy/jbossweb-tomcat55.sar/server.xml` under AJP 1.3 Connector. The default value is `8009`.)
5. Edit `$APACHE2_HOME/conf/httpd.conf` by adding the following to the LoadModules section:

   ```
   LoadModule jk_module modules/mod_jk.so
   Before Section 3:
   ```

```
#
# Mod_jk settings
#

    JkWorkersFile "conf/workers.properties"
    JkLogFile "logs/mod_jk.log"
    JkLogLevel info
    JkMount /cs/* jboss

# End of mod_jk settings
```

6. Test `httpd.conf` with the following:

   `cd $APACHE2_HOME/bin`
   `apachectl configtest`

   You will see a warning message and then "`Syntax OK`". Ignore the warning.

7. Start JBoss:

   `<jboss_install_dir>/bin/run.sh -c <instance name> -b <hostname>`

8. Start Apache:

   `$APACHE2_HOME/bin/apachectl start`

9. Load `http://<hostname>/cs/` to verify that the `/cs` directory is displayed.

Configuring JBoss for SSL through Apache 2.0.x

1. Generate a self-signed certificate:
   a. Edit `openssl.cnf` (usually in `/etc/ssl/`) by adding the following line:

      `dir = $APACHE2_HOME/demoCA`

   b. Set up the environment for the certificate authority certificate:

      ```
      cd $APACHE2_HOME
      mkdir demoCA
      cd demoCA
      mkdir certs
      mkdir crl
      touch index.txt
      mkdir newcerts
      echo "01" > serial
      mkdir private
      cd ..
      ```

   c. Generate the certificate authority key (must be done only the first time the certificate is created):

      `openssl genrsa -out ca.key 1024`

   d. Create a self-signed certificate authority certificate:
      1) Enter the command:
         `openssl req -new -x509 -key ca.key -out demoCA/cacert.pem`
      2) You will be prompted to fill in the following fields:

         `Country Name (2 letter code):`

```
State or Province Name (full name):
Locality Name (eg, city):
Organization Name (eg, company):
Organizational Unit Name (eg, section):
Common Name (eg, your name or your server's hostname):
Email Address:
```

e. Create the keystore by entering the following commands (shown in bold type) and filling in the fields (in quotes) with the information you used in the previous step:

```
keytool -genkey -alias serverapp -dname "cn=<common name>,
ou=<organizational unit>, o=<organization>, L=<locality>,
S=<state>, C=<country>" -storepass fatwire -keypass fatwire
-keystore newcerts
```

f. Export the keys for the keystore:

```
keytool -keystore newcerts -certreq -alias serverapp
-keypass fatwire -storepass fatwire -file serverapp.crs
```

g. Sign the exported key:

1) Enter the command:

```
openssl ca -in serverapp.crs -out serverapp.pem -keyfile
ca.key
```

2) When prompted to confirm the information you entered in the previous steps, enter "y" in the following fields:

```
Sign the certificate?
1 out of 1 certificate requests certified, commit?
```

h. Convert the keys to DER format:

```
openssl x509 -in serverapp.pem -out serverapp.der -outform
DER
```

i. Import the certificate authority certificate and the keys into the keystore:

1) Enter the command:

```
keytool -keystore newcerts -alias fatwirecs -keypass
fatwire -storepass fatwire -import -file demoCA/
cacert.pem
```

2) The certificate is displayed. Type **yes** next to "`Trust this certificate?`"

```
Valid from: Mon Apr 11 04:58:46 PDT 2005 until: Wed May
11 04:58:46 PDT 2005
Certificate fingerprints:
MD5:  80:B1:67:7A:46:17:3A:31:4D:23:38:57:47:19:2B:C5
SHA1:C7:AB:55:B3:9D:8F:DF:4A:BE:C2:48:11:8D:51:F8:17:
35:2E:4C:B3
Trust this certificate?:
```

2. Edit the `ssl.conf` file (located in `$APACHE2_HOME/conf/ssl.conf`) as shown below:
   a. Fill in the following fields:

   Note

   By default, the SSL port is `443`. If this server runs simultaneously with another Apache or Apache 2 server, the SSL port will need to be changed from `443`, to some other unused port.

   ```
   Listen <SSL_port>
   <VirtualHost _default_:<SSL_port>
   ServerName <hostname>:<SSL_port>
   ServerAdmin you@example.com
   ```

   b. Edit the following lines to point to the certificate and key files:

   ```
   SSLCertificateFile <$APACHE2_HOME>/demoCA/cacert.pem
   SSLCertificateKeyFile <$APACHE2_HOME>/ca.key
   ```

3. Restart Apache.
4. Access the following URL: `http://<hostname>:<SSL_port>/web-console/`
5. When prompted, accept the certificate.
6. Once you accept the certificate, the “JBoss Management Console” is displayed, confirming that JBoss is running over SSL.

Part 4

Content Server

This part shows you how to install Content Server, verify the installation, and complete any other post-installation steps that apply to your system. This part contains the following chapter:

- Chapter 5, “Installing and Configuring Content Server”

Chapter 5

Installing and Configuring Content Server

This chapter explains how to install and test Content Server on JBoss Application Server as a single-server or a clustered installation. This chapter contains the following sections:

- Installing Content Server
- Post-Installation Steps

Installing Content Server

After completing Steps I – IV.1 in the "Installation Quick Reference," on page 7, you install Content Server using the provided installer. The installation process consists of two stages.

In the first stage, the installer gathers necessary configuration information, installs the file structure, and deploys the CS application. At the end of the first stage, the installer displays an "Installation Actions" window describing the steps you must perform before proceeding to the second stage of the installation.

If you are using an Oracle database and require text attributes greater than 2000 characters, you must set the `cc.bigtext` property to `CLOB` after the CS application is deployed. (For instructions, see step 5 in the next section.)

If the first stage fails, the installer allows you to go back and modify your configuration options (except the database type), and retry the installation.

Note

If you need to change the type of database you have specified during the installation, you must delete the installed CS file structure and restart the installation.

In the second stage, the installer populates the database with the tables and data required for Content Server to function. If the second stage fails, the file structure and database tables must be deleted and the installation restarted from the beginning.

Running the Installer

To install Content Server

1. Make sure you have performed Steps I – IV.1 in the "Installation Quick Reference," on page 7.
2. Extract the Content Server installer archive into a temporary directory.
3. Change to the temporary directory containing the installer files.
4. Execute the installer script:
   - On Windows: **`csInstall.bat`**
   - On Unix: **`csInstall.sh`**

   The installer provides online help at each screen. Read the online help for detailed explanations of the options in each screen. If you encounter problems during the installation process, consult the online help for possible causes and solutions.
5. When the installer displays the "Installation Actions" pop-up window, do one or more of the following, depending on your configuration:
   - If you are installing on JBoss 4.2, do the following:
     1) Delete `commons-logging.jar` from `<jboss_install_dir>/server/<instance_name>/lib/`
     2) Move `commons-logging-1.1.jar` from `<jboss_install_dir>/server/<instance_name>/deploy/cs.war/WEB-INF/lib` to `<jboss_install_dir>/server/<instance_name>/lib`

- If you are using an Oracle database and require text attributes greater than 2000 characters, you must set the `cc.bigtext` property to `CLOB` as follows:
  1) Open the Property Editor by clicking the **Property Editor** button.
  2) In the Property Editor, open the `futuretense.ini` file.
  3) Click the **Database** tab.
  4) Locate the `cc.bigtext` property and set its value to `CLOB`.
  5) Save your changes and close the Property Editor.

6. When the installation completes successfully, perform the post-installation steps in the next section as required for your installation.

Post-Installation Steps

When the installation process completes successfully, perform the steps in this section, as required for your configuration:

A. Setting File Permissions (Unix Only)

If you installed Content Server on Unix, you must grant the "executable" permission to all files in the `<cs_install_dir>/bin` directory. To do so, perform the following steps:

1. Change to the `<cs_install_dir>/bin` directory.
2. Run the following command: **`chmod +x *`**
3. Restart JBoss.

B. Verifying the Installation

Verify the installation by logging in to Content Server as the administrator. This section covers the following types of installations:

- Web Installations
- Portal Installations

Web Installations

Logging in to the Advanced Interface

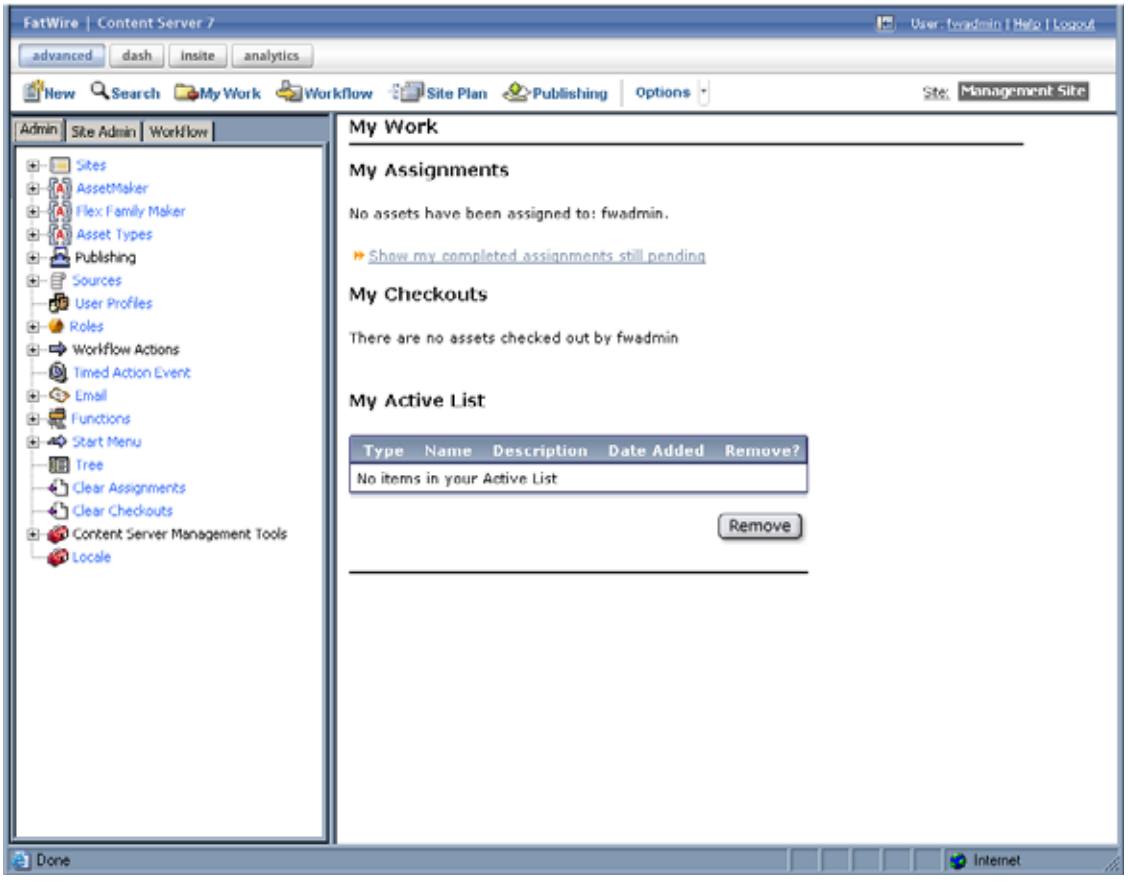
1. Point your browser to the following URL:

   `http://<hostname>:<port>/<context>/Xcelerate/LoginPage.html`

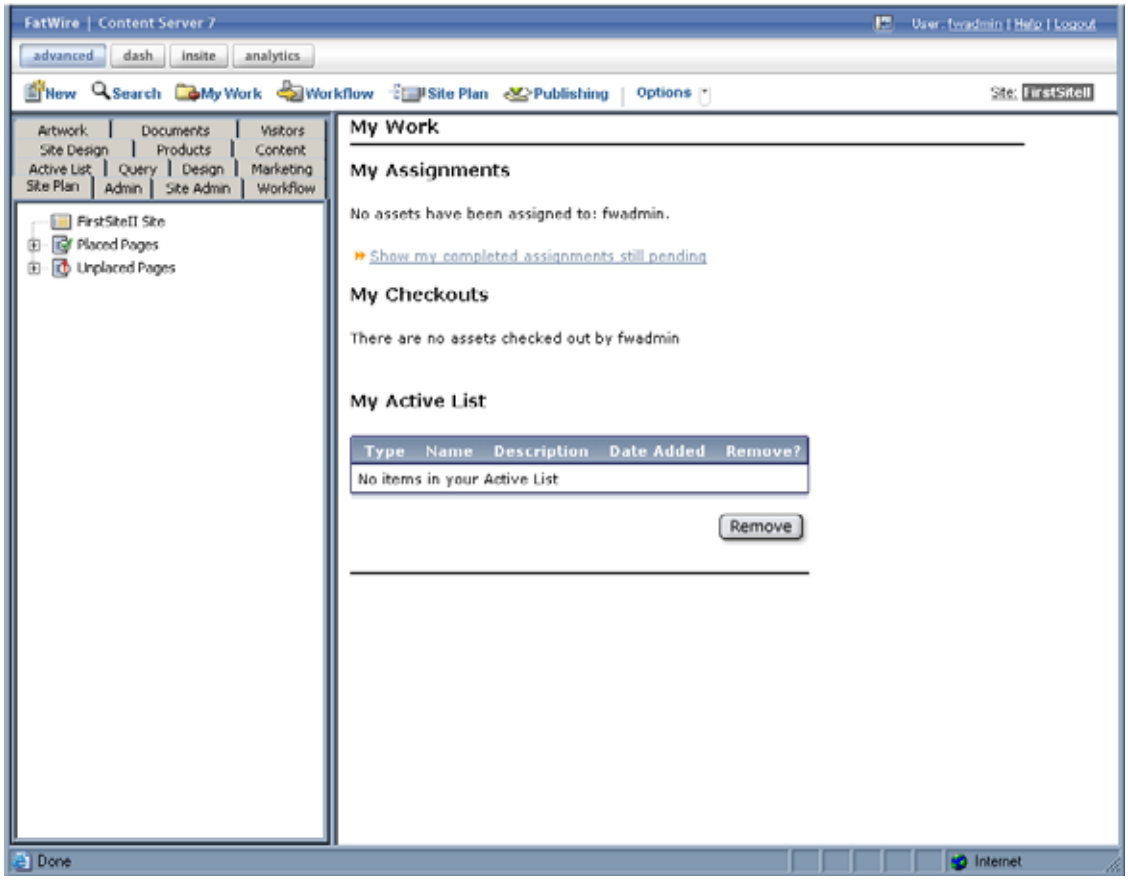
Content Server displays the Advanced interface login form:

2. Enter the following credentials:
   - User name: **fwadmin**
   - Password: **xceladmin**
3. Click **Login**.

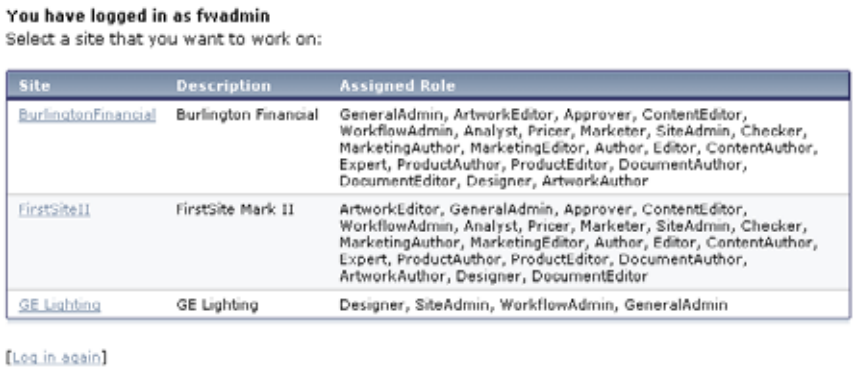
   Depending on whether you installed sample sites, one of the following happens:
   - If you did not install any sample sites, you are logged in to the built in Content Server management site. Only system administration functionality is available.



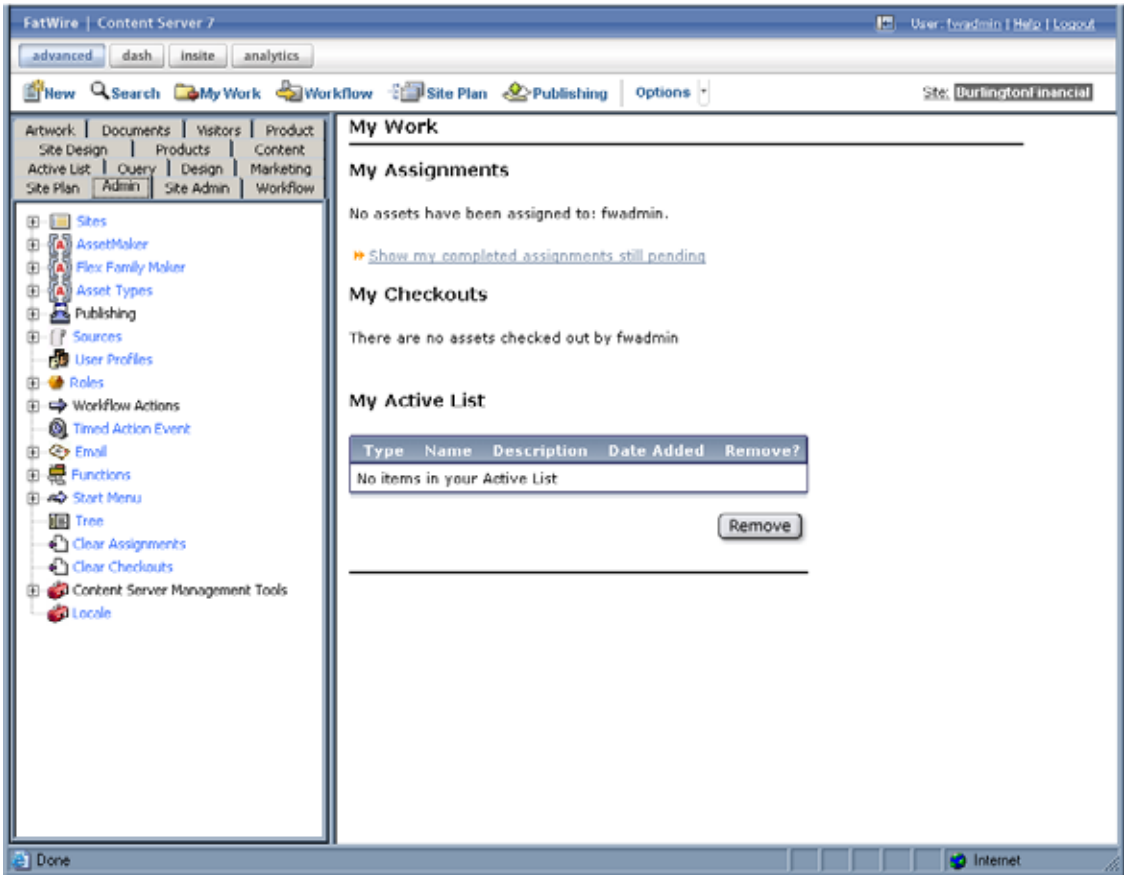
- If you installed one sample site, you are logged in to that site.



- If you installed more than one sample site, Content Server displays the "Select Site" screen. In such case, select the sample site you wish to log in to.



When you select a site, you are logged in to that site.



Logging in to the Dash Interface

1. Point your browser to the following URL:

   `http://<hostname>:<port>/<context>`

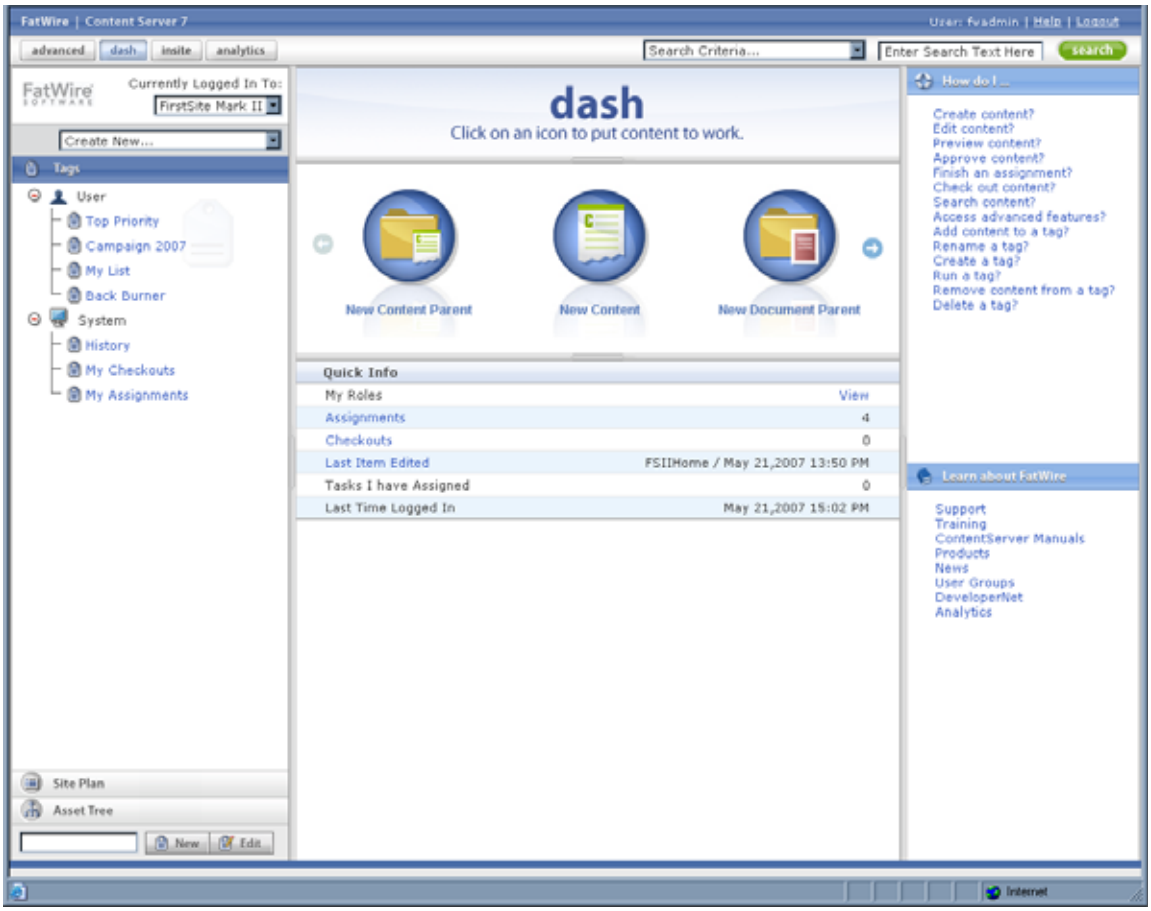
   Content Server displays the Dash interface login page.

2. Enter the following credentials:
   - User name: **fwadmin**
   - Password: **xceladmin**

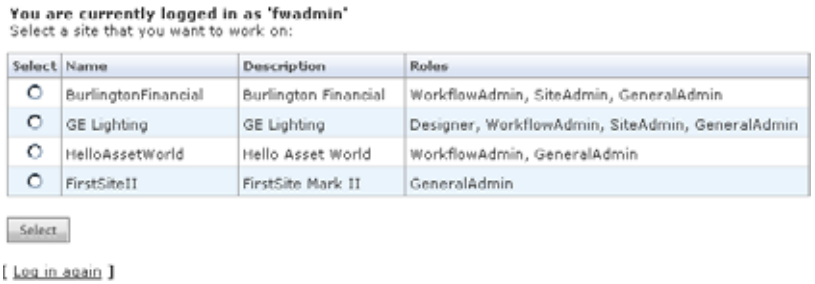
3. Click **Login**.

   Depending on whether you installed sample sites, one of the following happens:

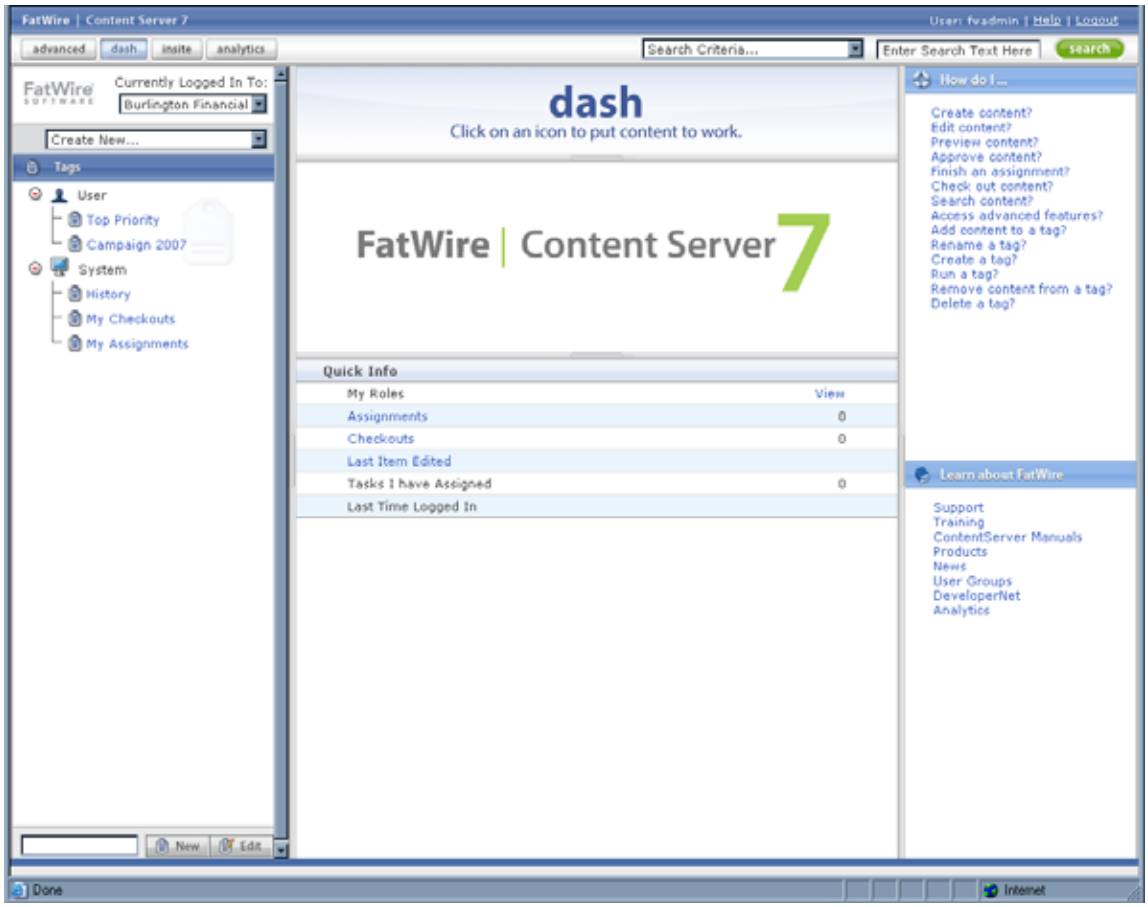
   - If you did not install any sample sites, Content Server displays a message notifying you of that fact. You will not be able to log in to the Dash interface until at least one site exists on your system.
   - If you installed one sample site, you are logged in to that site.



   - If you installed more than one sample site, Content Server displays the “Select Site” screen. In such case, select the sample site you wish to log in to.



When you select a site, you are logged in to that site.



Content Server is now ready for configuration. Follow the steps in the rest of this chapter, starting with "C. Integrating with LDAP (Required for Portal Installations)," on page 43.

Portal Installations

Verify the installation by accessing the portal interface in a browser:

```
http://<hostname>:<port>/portal
```

Content Server is now ready for configuration. Follow the steps in the rest of this chapter.

C. Integrating with LDAP (Required for Portal Installations)

LDAP integration is mandatory for portal installations, and optional for web installations. If you need to perform LDAP integration, you must do the following:

1. Set up a supported LDAP server of your choice. For instructions, see *Configuring Third-Party Software*.
2. Run the LDAP integration program included on the Content Server CD.

For more information, see the *LDAP Integration Guide*.

D. Configuring JBoss Portal for Content Server (Required for Portal Installations)

In this section, you will complete the following steps:

1. Configuring Content Server for LDAP Authorization
2. Setting Up the JBoss Portal Pages and Portlets

Note

The examples in this section are explicitly designed for OpenLDAP. For other LDAP server types, they must be modified to match the type

1. Configuring Content Server for LDAP Authorization

By default JBoss Portal uses its own database to authenticate users. However, for Content Server to work with JBoss Portal and allow for single sign-on, LDAP authorization must be configured as a fallback method. When authenticating, JBoss Portal will first refer to its own database, and then to the LDAP database.

Note

The default portal users are `admin` and `ldapuser`. The password for each is the same as the user name.

In the steps below, we assume that you will be connecting to OpenLDAP, but any LDAP server can be configured.

1. Change to the directory

   ```
   <jboss_portal_home>/server/<instance>/deploy/jboss-portal.sar/
      META-INF
   ```

   where `<instance>` is the name of the JBoss server instance (normally `default`), and `<jboss_portal_home>` is the directory that was created when you decompressed the `jboss-portal-2.6.1.zip` archive.

2. Make a backup copy of the `jboss-service.xml` file.
3. Edit the `jboss-service.xml` file and make the following changes:

   a. Locate the following section (located around line 539):

   ```
   <attribute name="ConfigFile">
   conf/identity/identity-config.xml
   </attribute>
   ```

   and change it to the following:

   ```
   <attribute name="ConfigFile">
   conf/identity/ldap_identity-config.xml
   </attribute>
   ```

   b. Save and close the file.

4. Change to the directory,

   `<jboss_portal_home>/server/<instance>/deploy/jboss-portal.sar/conf/identity`

   where `<instance>` is the name of the JBoss server instance (normally `default`), and `<jboss_portal_home>` is the directory that was created when you decompressed the `jboss-portal-2.6.1.zip` archive.

5. Make a backup copy of the `ldap_identity-config.xml` file.

> **Note**
>
> If you are using OpsenLDAP as your LDAP server, proceed to the next step to set up JBoss Portal for OpenLDAP. For instructions on setting up JBoss for other LDAP servers, see the *JBoss Portal 2.6.1-GA Reference Guide*. When you have set up your LDAP server, proceed to step 8 on page 49.

6. Edit the `ldap_identity-config.xml` file and replace its contents with the following code:

> **Note**
>
> In the code below, replace the values marked in bold red as follows:
>
> - Replace `password` with your LDAP password.
> - Replace `dc=fatwire,dc=com` with your domain and domain extension.

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
   ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
   ~~~~~~
  ~ JBoss, a division of Red Hat
   ~
  ~ Copyright 2006, Red Hat Middleware, LLC, and individual
   ~
  ~ contributors as indicated by the @authors tag. See the
   ~
  ~ copyright.txt in the distribution for a full listing of
   ~
  ~ individual contributors.
   ~
  ~                                                                      ~
  ~ This is free software; you can redistribute it and/or modify it
   ~
  ~ under the terms of the GNU Lesser General Public License as
   ~
  ~ published by the Free Software Foundation; either version 2.1 of
   ~
  ~ the License, or (at your option) any later version.
   ~
  ~                                                                      ~
```

```
  ~ This software is distributed in the hope that it will be useful,
    ~
  ~ but WITHOUT ANY WARRANTY; without even the implied warranty of
    ~
  ~ MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU
    ~
  ~ Lesser General Public License for more details.
    ~
  ~                                                                    ~
  ~ You should have received a copy of the GNU Lesser General Public
    ~
  ~ License along with this software; if not, write to the Free
    ~
  ~ Software Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA
    ~
  ~ 02110-1301 USA, or see the FSF site: http://www.fsf.org.
    ~

    ~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
    ~~~~~~~~->

    <!--<!DOCTYPE identity-configuration PUBLIC
    "-//JBoss Portal//DTD JBoss Identity Configuration 1.0//EN"
    "http://www.jboss.org/portal/dtd/identity-config_1_0.dtd">-->

<identity-configuration>
   <datasources>
      <datasource>
         <name>LDAP</name>
         <config>
            <option>
               <name>host</name>
               <value>127.0.0.1</value>
            </option>
            <option>
               <name>port</name>
               <value>389</value>
            </option>
            <option>
               <name>adminDN</name>
               <value>cn=Manager,dc=fatwire,dc=com</value>
            </option>
            <option>
               <name>adminPassword</name>
               <value>password</value>
            </option>
         </config>
      </datasource>
```

```
</datasources>
<modules>
   <module>
      <!--type used to correctly map in IdentityContext registry-->
      <type>User</type>
      <implementation>LDAP</implementation>
      <config/>
   </module>
   <module>
      <type>Role</type>
      <implementation>LDAP</implementation>
      <config/>
   </module>
   <module>
      <type>Membership</type>
      <implementation>LDAP</implementation>
      <config/>
   </module>
   <module>
      <type>UserProfile</type>
      <implementation>DELEGATING</implementation>
      <config>
         <option>
            <name>ldapModuleJNDIName</name>
            <value>java:/portal/LDAPUserProfileModule</value>
         </option>
      </config>
   </module>
   <module>
      <type>DBDelegateUserProfile</type>
      <implementation>DB</implementation>
      <config>
         <option>
            <name>randomSynchronizePassword</name>
            <value>true</value>
         </option>
      </config>
   </module>
   <module>
      <type>LDAPDelegateUserProfile</type>
      <implementation>LDAP</implementation>
      <config/>
   </module>
</modules>
```

```
<options>
   <option-group>
      <group-name>common</group-name>
      <option>
         <name>userCtxDN</name>
         <value>ou=People,dc=fatwire,dc=com</value>
      </option>
      <option>
         <name>uidAttributeID</name>
         <value>cn</value>
      </option>
      <option>
         <name>passwordAttributeID</name>
         <value>userPassword</value>
      </option>

      <option>
         <name>roleCtxDN</name>
         <value>ou=Groups,dc=fatwire,dc=com</value>
      </option>
      <option>
         <name>ridAttributeId</name>
         <value>cn</value>
      </option>
      <option>
         <name>roleDisplayNameAttributeID</name>
         <value>cn</value>
      </option>
      <option>
         <name>membershipAttributeID</name>
         <value>uniqueMember</value>
      </option>
      <option>
         <name>membershipAttributeIsDN</name>
         <value>true</value>
      </option>
   </option-group>
   <option-group>
      <group-name>userCreateAttibutes</group-name>
      <option>
         <name>objectClass</name>
         <!--This objectclasses should work with Red Hat Directory-->
         <value>top</value>
         <value>person</value>
         <value>organizationalPerson</value>
```

```
         </option>
         <!--Schema requires those to have initial value-->
         <option>
            <name>cn</name>
            <value>none</value>
         </option>
         <option>
            <name>sn</name>
            <value>none</value>
         </option>
      </option-group>

      <option-group>
         <group-name>roleCreateAttibutes</group-name>
         <option>
            <name>objectClass</name>
            <value>top</value>
            <value>groupOfUniqueNames</value>
         </option>
         <!--Schema requires those to have initial value-->
         <option>
            <name>cn</name>
            <value>none</value>
         </option>
         <!--Some directory servers require this attribute to be valid
   DN-->
         <!--For safety reasons point to the admin user here-->
         <option>
            <name>member</name>
            <value>cn=admin,ou=People,dc=fatwire,dc=com</value>
         </option>
      </option-group>
   </options>
</identity-configuration>
```

7. Save and close the file.
8. Create an LDIF file containing the following code, and import it into your LDAP server.

Note

If you are using an LDAP server other than OpenLDAP, you will need to modify the code below as required by your LDAP server before proceeding. For instructions, see your LDAP server documentation.

a. Create a file named `jboss-config.ldif` and paste the following code into it:

> **Note**
>
> In the code below, replace the values marked in bold red as follows:
>
> - For each default user, replace `{SSHA}password` with a password of your choice. (If you are using OpenLDAP, you can change user passwords using the `slappasswd` command.)
> - Replace `dc=fatwire,dc=com` with your domain and domain extension.

```
dn: cn=admin,ou=People, dc=fatwire,dc=com
userPassword: password
objectClass: top
objectClass: organizationalPerson
objectClass: person
sn: admin
cn: admin

dn: cn=ldapuser,ou=People, dc=fatwire,dc=com
userPassword: password
description: JBoss Portal ldapuser User for test LDAP
integration password=username
objectClass: top
objectClass: organizationalPerson
objectClass: person
sn: ldapuser
cn: ldapuser

dn: cn=User,ou=Groups, dc=fatwire,dc=com
description: the JBoss Portal user group
objectClass: top
objectClass: groupOfUniqueNames
uniqueMember: cn=ldapuser,ou=People,dc=fatwire,dc=com
cn: User

dn: cn=Admin,ou=Groups, dc=fatwire,dc=com
description: the JBoss Portal admin group
objectClass: top
objectClass: groupOfUniqueNames
uniqueMember: cn=admin,ou=People,dc=fatwire,dc=com
cn: Admin
```

b. Save and close the file.

c. Import the jboss-config.ldif file into your LDAP server. For instructions, refer to your LDAP server's documentation.

For example, to import the file into OpenLDAP, use the following command:

```
ldapadd -D 'cn=Manager,dc=fatwire,dc=com' -w <password> -f
   jboss-config.ldif
```

9. Edit the file `<jboss_home>/bin/run.conf` and make the following changes:
   a. Locate the following section:

```
if [ "x$JAVA_OPTS" = "x" ]; then
   JAVA_OPTS="-Xms128m -Xmx512m
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000"
fi
```

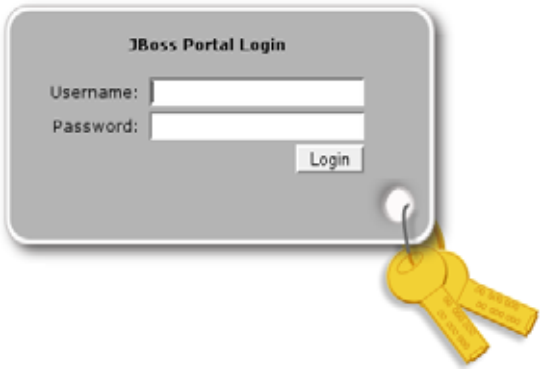
   b. Replace the section with the following code:

```
if [ "x$JAVA_OPTS" = "x" ]; then
  JAVA_OPTS="-Xms256m -Xmx512m
-Dsun.rmi.dgc.client.gcInterval=3600000
-Dsun.rmi.dgc.server.gcInterval=3600000
-XX:MaxPermSize=128m"
fi
```

   c. Save and close the file.
10. Restart JBoss Portal.
11. Test your LDAP integration by logging in to the portal as the `admin` and `ldapuser` users:
   a. Point your browser to the following URL:

      `http://<hostname>:8080/portal`

   b. In the upper right corner of the page, click **Login** to access the login page.



   c. In the **Username** field, enter `admin`.
   d. In the **Password** field, enter the `admin` user's password.
   e. Click **Login**.

      The default portal page appears with the **Dash** and **Admin** links now visible in the upper right corner.

   f. Log out of the portal and repeat steps a–e for the `ldapuser` user. When you log in as the `ldapuser` user, you will see the **Dash** link but not the **Admin** link in the upper right corner of the page.

12. Continue on to the next section, "Setting Up the JBoss Portal Pages and Portlets," on page 53, to set up portal pages and portlets.

Note

In order to allow CS users to access the portal, you must add them to either the `user` or `admin` LDAP groups (these groups are created by the Content Server LDAP integrator pogram). In a staging environment, we suggest that you add the `fwadmin` and `firstsite` users to the `admin` group, and all other users to the `user` group.

2. Setting Up the JBoss Portal Pages and Portlets

In the steps that follow, you will add pages on which to display Content Server's portlets. You will then enable the portlets and place them on the pages. Default pages and portlets are named as shown in Table 3.

Figure 1: "Spark Display" page

Table 3: Default Page and Portlet Names

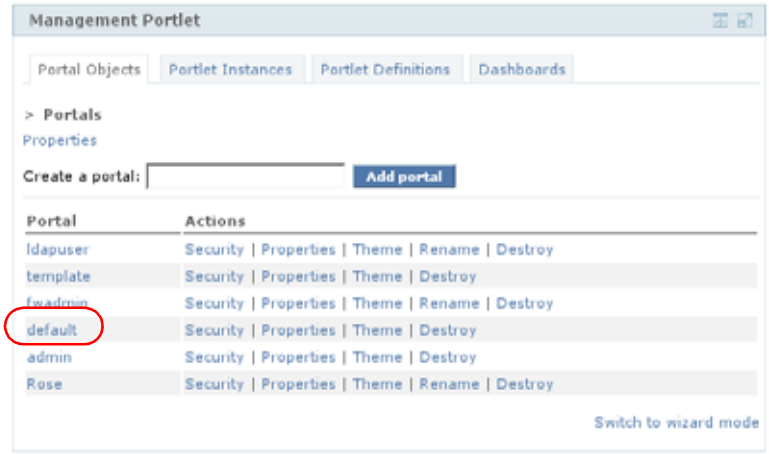
Page Name	Portlet Name	Page Name	Portlet Name
CS Admin (optional, as it duplicates the CS administrator interface)	ClearAssignments ClearCheckouts ContentDefinition PublishTarget Roles UserPortlet	CS Documents	ActiveDocuments CheckedOutDocuments DocumentAssignments DocumentHistory MyDocuments SearchDocuments SiteInfo
CS Content	ActiveContent CheckedOutContent ContentAssignments ContentHistory CreateContent PublishConsole SearchContent Site Info	Spark Display (optional; cannot be created unless you installed the "Spark" sample site)	SparkAds SparkDocuments SparkNews SparkJobs

1. Log in to the portal as the `admin` user:
   a. Point your browser to the following URL:
      `http://<server>:<port>/portal`
   b. In the upper right corner of the page, click **Login**.
2. In the login screen, enter the user name and password of the `admin` user, then click **Login**.
3. In the upper right corner of the page, click **Admin**.

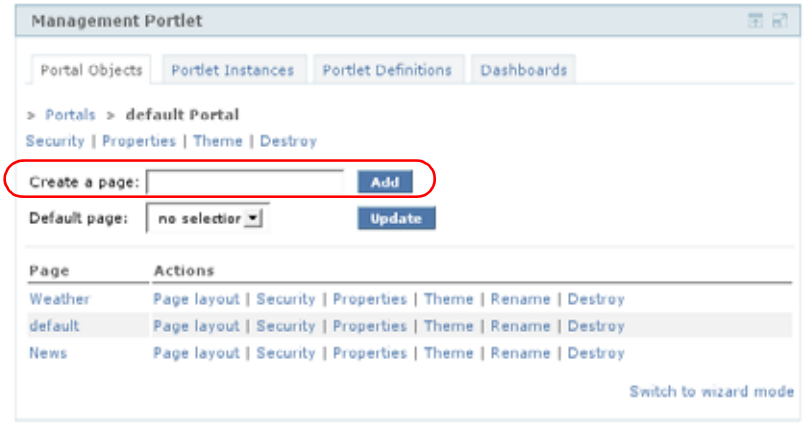
Note

The next steps show you how to create a new page, how to set the page style, and how to add portlets to the page. Our example is based on four Spark sample site user portlets; you use the same steps to configure the admin portlets.

4. Click the link representing the portal you wish to edit. (We are assuming you are editing the **default** portal.)

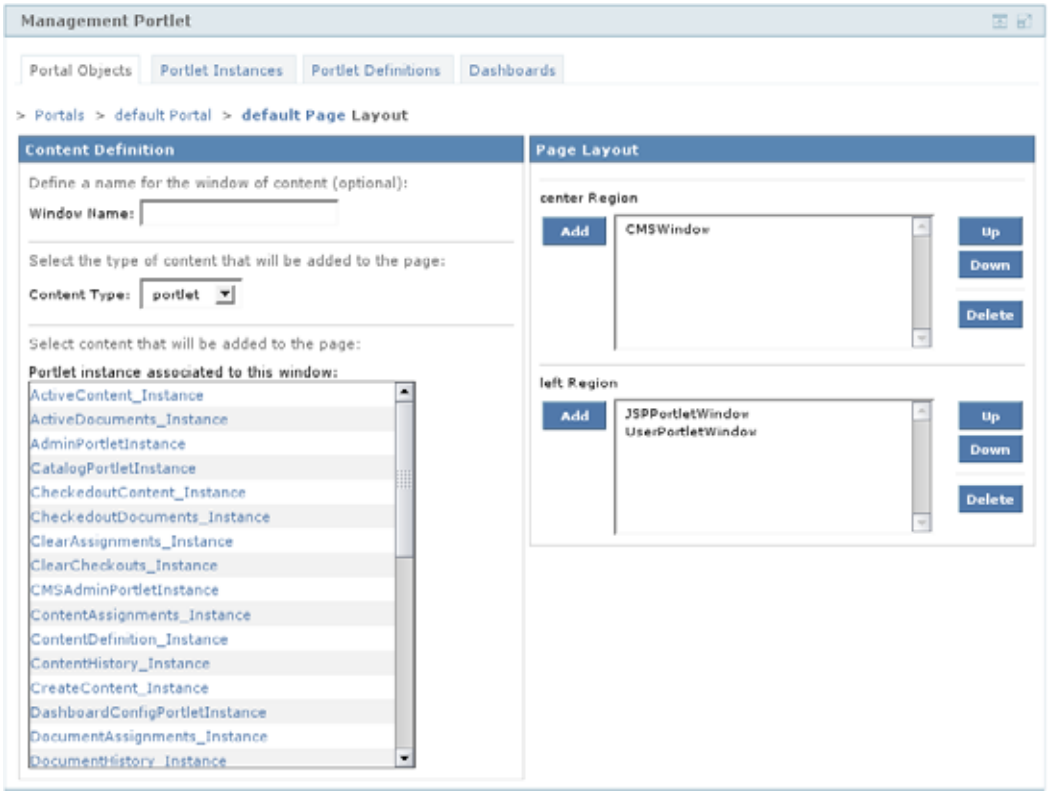


5. Create the “Spark Samples” page. In the **Create a page** field, enter **Spark Samples** and click **Add**.



The “Spark Samples” page you just created appears in the list.

6. Add the required portlets to the “Spark Samples” page you just created:
   a. In the “Spark Samples” page row in the list, click **Page Layout**.
   The following screen appears:



   b. In the “Portlet instance associated to this window” list, select **SparkAds_Instance**.
   The page refreshes and the selected entry is now highlighted.
   c. In the “Left Region” list, click **Add**.

> **Note**
>
> If you wish to deploy the portlet to another region, click the **Add** button next to that region.

d. Repeat steps a–c to add the **SparkDocuments_Instance**, **SparkNews_Instance** and **SparkJobs_Instance** portlets to the page as follows:

- Add the “SparkDocuments” and “SparkNews” portlets to the center region.
- Add the “SparkJobs” portlet to the left region.

When you are finished, your portal should look similar to the following:

Note

The exact portlets you need to add will vary depending on your installation; for details, consult your site developers.

7. Repeat steps 5 and 6 to add and populate the remaining pages, as described in Table 3, on page 53.

8. Test your configuration:

a. Log out of the portal by clicking **Logout** in the upper right corner of the page.

The default portal page should now contain a tab for each page you have added.

b. Click each tab to see whether you can open the corresponding page. If you cannot open a page, check to see that you have completed the steps in this section correctly.

E. Setting Up a Content Server Cluster (Optional)

Before beginning the steps in this section, make sure of the following:

- You are installing a vertical cluster (JBoss application server or portal server instances are installed on the same machine).
- You have already installed and configured an instance of JBoss, and it is running a verified Content Server system.

If the conditions above are satisfied, complete the following steps in this section:

1. Creating Cluster Members

1. Create a new JBoss instance and Content Server installation directory by completing "Step II. Setting Up the Environment," on page 20.
2. Ensure that the port settings of the new instance are unique across the cluster members. Follow the steps in "Step III. Setting Up Multiple Instances of JBoss to Run Simultaneously," on page 24.
3. Run the Content Server installer on the newly created JBoss instance (by executing either **`csInstall.bat`** or **`csInstall.sh`**). Perform the steps exactly as you performed them when installing the first Content Server instance, but with the following exceptions:
   a. In the "Installation Directory" screen, select the Content Server directory that you created in step 1 of this section.
   b. In the "Clustering" screen, select **Cluster Member**.
   c. In the "Content Server Shared File System" screen, enter the path to the shared file system of the primary cluster member.
   d. In the "Content Server URL Parameters" screen, use the IP address and port that you set in step 2 for this application server instance.
   e. In the "Content Server Application Deployment" screen, enter the path to the JBoss instance you created in step 1 of this section.
4. Ensure there is a `usedisksync` directory in the `Shared` directory of your primary CS installation (the first member that was created). If `usedisksync` is not already there, then create it with the following command:

   `mkdir <cs_install_dir>/Shared/usedisksync`

   Note

   To complete the next step, make sure your X-Display variable is set.

5. To finalize the clustering process, edit the following properties **for each cluster member** using the Property Editor:
   a. Run the Property Editor, `propeditor.sh` (or `.bat`), in `<cs_install_dir>`.
   b. Click **File** > **Open**, and open the `futuretense.ini file` in your Content Server installation directory.

c. Click **Cluster**.

Three properties should be displayed on the **Items** tab:

- `cc.cacheNoSync`: Assign a value of `false`.
- `ft.sync`: Assign the same value to all members of this cluster. **Ex:** cluster1
- `ft.usedisksync`: Set its value to the path to the `usedisksync` directory created in step 4.

6. Verify the newly created cluster member. For instructions, see "B. Verifying the Installation," on page 37.
7. To create additional cluster members, repeat the steps in this section. Once the cluster is created, continue with the next step, "2. Balancing Load with mod_jk."

2. Balancing Load with mod_jk

1. Open `<jboss_install_dir>/server/<instance_name>/deploy/jbossweb-tomcat55.sar/server.xml` and add the following attribute to the Engine tag:

```
jvmRoute="node1"
```

> **Note**
>
> A cluster member can be given any node name as long as the names are unique across members.

2. In step 4 on page 28, make sure the `workers.properties` in `$APACHE2_HOME/conf` has the following content:

```
ps=/
worker.list=node1, node2, loadbalancer

worker.node1.port=<ajp port1>
worker.node1.host=<hostname>
worker.node1.type=ajp13
worker.node1.lbfactor=1
worker.node1.cachesize=1

worker.node2.port=<ajp port2>
worker.node2.host=<hostname>
worker.node2.type=ajp13
worker.node2.lbfactor=1
worker.node2.cachesize=1
```

```
worker.loadbalancer.type=lb
worker.loadbalancer.balanced_workers=node1,node2
/cs/*=loadbalancer
```

Note

This configuration supports two cluster members and one load balancer. For each additional cluster member, add the member name to the `worker.list`, `worker.loadbalancer.balanced_workers`, and add the following lines:

```
worker.<member name>.port=<ajp port>
worker.<member name>.host=<hostname>
worker.<member name>.type=ajp13
worker.<member name>.lbfactor=1
worker.node1.cachesize=1
```

Use the names stored in the `server.xml` file mentioned in step 1 of this section.

3. In step 5 on page 28, edit line `JkMount /cs/* jboss` to read:
   `JkMount /cs/* loadbalancer`
4. Restart Apache for the changes to take effect.

F. Setting Up Content Server for Its Business Purpose

Once you have completed the Content Server installation, you are ready to configure it for business use. For instructions, see the *Content Server Administrator's Guide* and the *Content Server Developer's Guide*. The guides explain how to create and enable a content management environment including the data model, content management sites, site users, publishing functions, and client interfaces.