

# FatWire | Content Server 7

Version 7.0.2

## Installing Content Server with Tomcat Application Server

**Document Revision Date:** Jun. 15, 2011



FATWIRE CORPORATION PROVIDES THIS PUBLICATION "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. In no event shall FatWire be liable for any loss of profits, loss of business, loss of use of data, interruption of business, or for indirect, special, incidental, or consequential damages of any kind, even if FatWire has been advised of the possibility of such damages arising from this publication. FatWire may revise this publication from time to time without notice. Some states or jurisdictions do not allow disclaimer of express or implied warranties in certain transactions; therefore, this statement may not apply to you.

Copyright © 2007–2011 FatWire Corporation. All rights reserved.

This product may be covered under one or more of the following U.S. patents: 4477698, 4540855, 4720853, 4742538, 4742539, 4782510, 4797911, 4894857, 5070525, RE36416, 5309505, 5511112, 5581602, 5594791, 5675637, 5708780, 5715314, 5724424, 5812776, 5828731, 5909492, 5924090, 5963635, 6012071, 6049785, 6055522, 6118763, 6195649, 6199051, 6205437, 6212634, 6279112 and 6314089. Additional patents pending.

FatWire, Content Server, Content Server Bridge Enterprise, Content Server Bridge XML, Content Server COM Interfaces, Content Server Desktop, Content Server Direct, Content Server Direct Advantage, Content Server DocLink, Content Server Engage, Content Server InSite Editor, Content Server Satellite, and Transact are trademarks or registered trademarks of FatWire Corporation in the United States and other countries.

iPlanet, Java, J2EE, Solaris, Sun, and other Sun products referenced herein are trademarks or registered trademarks of Sun Microsystems, Inc. AIX, IBM, WebSphere, and other IBM products referenced herein are trademarks or registered trademarks of IBM Corporation. WebLogic is a registered trademark of BEA Systems, Inc. Microsoft, Windows and other Microsoft products referenced herein are trademarks or registered trademarks of Microsoft Corporation. UNIX is a registered trademark of The Open Group. Any other trademarks and product names used herein may be the trademarks of their respective owners.

This product includes software developed by the Apache Software Foundation (<http://www.apache.org/>) and software developed by Sun Microsystems, Inc. This product contains encryption technology from Phaos Technology Corporation.

You may not download or otherwise export or reexport this Program, its Documentation, or any underlying information or technology except in full compliance with all United States and other applicable laws and regulations, including without limitations the United States Export Administration Act, the Trading with the Enemy Act, the International Emergency Economic Powers Act and any regulations thereunder. Any transfer of technical data outside the United States by any means, including the Internet, is an export control requirement under U.S. law. In particular, but without limitation, none of the Program, its Documentation, or underlying information of technology may be downloaded or otherwise exported or reexported (i) into (or to a national or resident, wherever located, of) Cuba, Libya, North Korea, Iran, Iraq, Sudan, Syria, or any other country to which the U.S. prohibits exports of goods or technical data; or (ii) to anyone on the U.S. Treasury Department's Specially Designated Nationals List or the Table of Denial Orders issued by the Department of Commerce. By downloading or using the Program or its Documentation, you are agreeing to the foregoing and you are representing and warranting that you are not located in, under the control of, or a national or resident of any such country or on any such list or table. In addition, if the Program or Documentation is identified as Domestic Only or Not-for-Export (for example, on the box, media, in the installation process, during the download process, or in the Documentation), then except for export to Canada for use in Canada by Canadian citizens, the Program, Documentation, and any underlying information or technology may not be exported outside the United States or to any foreign entity or "foreign person" as defined by U.S. Government regulations, including without limitation, anyone who is not a citizen, national, or lawful permanent resident of the United States. By using this Program and Documentation, you are agreeing to the foregoing and you are representing and warranting that you are not a "foreign person" or under the control of a "foreign person."

*Installing Content Server with Tomcat Application Server*

Document Revision Date: Jun. 15, 2011

Product Version: 7.0.2

**FatWire Technical Support**

[www.fatwire.com/Support](http://www.fatwire.com/Support)

**FatWire Headquarters**

FatWire Corporation  
330 Old Country Road  
Suite 303  
Mineola, NY 11501  
[www.fatwire.com](http://www.fatwire.com)

## Table of Contents

<b>1 Introduction</b> .....	<b>5</b>
About This Guide .....	6
How This Guide Is Organized .....	6
Terms and Acronyms .....	6
Graphics in This Guide .....	6
Installation Quick Reference .....	7

### Part 1. Database

<b>2 Setting Up a Database</b> .....	<b>13</b>
--------------------------------------	-----------

### Part 2. Application Server

<b>3 Installing and Configuring Tomcat</b> .....	<b>17</b>
Start/Stop Commands .....	18
Step I. Installing Tomcat .....	18
Step II. Configuring Tomcat .....	19
Setting Up Directories for Content Server and Tomcat .....	19
Setting Up Multiple Instances of Tomcat to Run Simultaneously .....	25
Configuring Tomcat for SSL .....	26

### Part 3. Web Server

<b>4 Integrating with Apache Web Server</b> .....	<b>29</b>
Step I. Install Apache Web Server 2.0.x .....	30
Step II. Integrate Tomcat with Apache Web Server .....	30

## Part 4. Content Server

<b>5</b>	<b>Installing and Configuring Content Server</b> .....	<b>35</b>
	Installing Content Server .....	36
	Running the Installer .....	36
	Post-Installation Steps .....	37
	A. Setting File Permissions (Unix Only) .....	37
	B. Adding the CS Binaries Directory to the Library Path Variable .....	37
	C. Verifying the Installation .....	38
	D. Integrating with LDAP (Optional) .....	43
	E. Setting Up a Content Server Cluster (Optional) .....	43
	F. Setting Up Content Server for Its Business Purpose .....	46

## Chapter 1

# Introduction

This document provides guidelines for installing Content Server on Tomcat Application Server 5.5/6, 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](#)
- [Installation Quick Reference](#)

## About This Guide

This guide covers the installation, configuration, and maintenance of Tomcat Application Server 5.5/6, as required to support Content Server. This includes configuration of one or more Tomcat instances, creation of backend databases, and integration of Tomcat with Apache Web Server. This guide also provides instructions for installing Content Server in single-member and vertically clustered environments.

This guide does not cover the following topics, as they fall outside the scope of this guide:

- Tomcat SSL configuration for simultaneously run instances
- LDAP integration. For information about this topic, see the *LDAP Integration Guide*.

## 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 “[Installation Quick Reference](#),” on [page 7](#)).

## Terms and Acronyms

<code>&lt;tomcat_home&gt;</code>	Path to the Tomcat installation directory ( <code>\$CATALINA_HOME</code> )
<code>&lt;cs_install_dir&gt;</code>	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 database, application server, and optionally the web server that supports 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 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 steps below 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

To set up Tomcat Application Server, complete the steps in [Chapter 3, “Installing and Configuring Tomcat”](#):

1. [“Installing Tomcat,” on page 18.](#)
2. [“Setting Up Directories for Content Server and Tomcat,” on page 19](#) for Content Server and for the Tomcat instance(s) on which Content Server will be running.

## Note

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

You also have the option to integrate your installation with an LDAP server. 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 Tomcat 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 Tomcat instance runs on a unique port number. For instructions, see [“Setting Up Multiple Instances of Tomcat to Run Simultaneously,” on page 25.](#)
4. If you plan on running a single instance over SSL, complete the steps in [“Configuring Tomcat for SSL,” on page 26.](#)

### III. Integrate Tomcat with Apache Web Server

If you are planning to use Apache web server or you are creating a vertical cluster that is load balanced through the web server, complete the following steps (in [Chapter 4](#)):

[Step I. Install Apache Web Server 2.0.x \(page 30\)](#)

[Step II. Integrate Tomcat with Apache Web Server \(page 30\)](#)

### IV. Install and Configure Content Server

In this step, you will 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).

1. Before installing Content Server, do the following:
  - a. Start a Tomcat 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 set the `cc.bigtext` property to `CLOB` after the CS application is deployed half-way through the installation. For instructions, see [step 5](#) 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. Add the CS binaries directory to your library path variable as explained in [“Adding the CS Binaries Directory to the Library Path Variable,” on page 37](#).
  - c. Verify the Content Server installation by logging in as the administrator. For instructions, see [“Verifying the Installation,” on page 38](#).
  - d. If you want to perform LDAP integration, follow the steps in [“Integrating with LDAP \(Optional\),” on page 43](#).
  - e. If you are creating a vertically clustered system, follow instructions in [“Setting Up a Content Server Cluster \(Optional\),” on page 43](#).

- 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 the Tomcat Application Server as well as integrating Tomcat with Apache Web Server.

This part contains the following chapter:

- [Chapter 3, “Installing and Configuring Tomcat”](#)



## Chapter 3

# Installing and Configuring Tomcat

The chapter shows you how to install and configure Tomcat Application Server for Content Server, for running over SSL, and for simultaneously running multiple instances of Tomcat.

This is not an exhaustive chapter, as it covers the installation of Tomcat Application Server only so far as needed to install and run Content Server. For more extensive documentation on the installation process, see the documentation on the Apache Tomcat website.

This chapter contains the following sections:

- [Start/Stop Commands](#)
- [Installing Tomcat](#)
- [Configuring Tomcat](#)

## Start/Stop Commands

This section lists commands for starting and stopping Tomcat instances.

### Note

All commands require that `CATALINA_HOME`, `CATALINA_BASE`, and `JAVA_HOME` are set to the proper directories. To specify which instance to start or shut down, the `CATALINA_BASE` must be set to the corresponding Tomcat instance directory.

Action	Command
Start the Tomcat instance:	<code>\$CATALINA_HOME/bin/startup.sh</code>
Shut down the Tomcat instance:	<code>\$CATALINA_HOME/bin/shutdown.sh -force</code>

## Step I. Installing Tomcat

### Note

We assume that you are using a UNIX system.

### To install the Tomcat Application Server

1. Create an installation directory for Tomcat.
2. Download and install JDK 1.5.x from Sun:
3. Set `CATALINA_HOME` as the Tomcat installation directory export:
 

```
CATALINA_HOME=<path_to_tomcat_home>
```
4. Decompress the Tomcat archive file into the Tomcat installation directory:
 

```
gzip -d apache-tomcat-5.5.16.tar.gz
tar -xvf apache-tomcat-5.5.16.tar
```
5. Edit `$CATALINA_HOME/bin/catalina.sh` by adding the following line after the first comment block:
 

```
CATALINA_PID="$CATALINA_HOME"/tomcat.pid
```

 This ensures that the Tomcat process is killed when the `shutdown.sh` command is executed with the `-force` argument.
6. Set the `JAVA_HOME` variable to the JDK folder of the version of Java that will be used. For example:
 

```
export JAVA_HOME=/opt/jdk/jdk1.5.0_06
```
7. Start the application server by running the startup command.
 

```
$CATALINA_HOME/bin/startup.sh
```
8. Access the following URL in a web browser: `http://<hostname>:8080/`  
This brings you to Tomcat's default homepage.

9. Shut down the application server by running the `shutdown` command.

```
$CATALINA_HOME/bin/shutdown.sh -force
```

10. Edit the `$CATALINA_HOME/conf/tomcat-users.xml` file by adding the following lines:

```
<user username="admin" password="<password>"
      roles="admin,manager,tomcat"/>
```

This creates a new user with the permissions to access both the Admin and Manager tools.

11. After you have successfully installed Tomcat, continue with the next section, [“Configuring Tomcat.”](#)

## Step II. Configuring Tomcat

Complete the steps in this section as required for your installation:

- [Setting Up Directories for Content Server and Tomcat](#)
- [Setting Up Multiple Instances of Tomcat to Run Simultaneously](#)
- [Configuring Tomcat for SSL](#)

### Setting Up Directories for Content Server and Tomcat

1. Create a directory for Content Server and change to the directory:

```
mkdir CS7.0
cd CS7.0
```

2. In this directory create two subdirectories:

- a. Fatwire – The directory where Content Server will be installed:

```
mkdir Fatwire
```

- b. `cs_tomcat` – The directory where the Tomcat instance for Content Server will reside:

```
mkdir cs_tomcat
```

3. Set `CATALINA_BASE` as the new Tomcat instance directory:

```
export CATALINA_BASE=<path to cs_tomcat directory>
```

4. In the `cs_tomcat` directory create the following directories: `conf`, `logs`, `temp`, `webapps`, and `work`:

```
cd cs_tomcat
mkdir conf
mkdir logs
mkdir temp
mkdir webapps
mkdir work
```

5. In the `webapps` directory, create a directory named `cs`:

```
cd webapps
mkdir cs
```

6. Copy `tomcat.gif` from `$CATALINA_HOME/webapps/ROOT` to `$CATALINA_BASE/webapps/cs`:

```
cd cs
cp $CATALINA_HOME/webapps/ROOT/tomcat.gif $CATALINA_BASE/webapps/cs
```

This image will be used to test whether the new instance is set up properly.

Here is an example directory structure:

```
CS7.0/
  cs_tomcat/
    conf/
    logs/
    temp/
    webapps/
      cs/
        tomcat.gif
  work/
  Fatwire/
```

7. Copy the `catalina.policy`, `tomcat-users.xml` and `web.xml` files from `$CATALINA_HOME/conf` to `$CATALINA_BASE/conf`
- ```
cd $CATALINA_HOME/conf
cp catalina.policy tomcat-users.xml web.xml $CATALINA_BASE/conf
```
8. Choose the data source information that corresponds to the database you will be using and place the required files in `$CATALINA_HOME/common/lib`. This data will be inserted into the `server.xml` file in the next step.

**Table 1:** Data Source Information

| Database Driver                  | Parameters          | Values                                                                                                                                                         |
|----------------------------------|---------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>JTDS (third-party driver)</b> | DriverClass         | <code>net.sourceforge.jtds.jdbcx.JtdsDataSource</code>                                                                                                         |
|                                  | Required .jar files | <code>jtds-1.2.jar</code>                                                                                                                                      |
|                                  | URL                 | <code>jdbc:jtds:sqlserver://&lt;server&gt;:&lt;dbport&gt;/&lt;dbname&gt;</code><br><br><b>Ex:</b><br><code>jdbc:jtds:sqlserver://10.120.14.22:1433/CS70</code> |

**Table 1:** Data Source Information (*continued*)

| Database Driver           | Parameters          | Values                                                                                                                  |
|---------------------------|---------------------|-------------------------------------------------------------------------------------------------------------------------|
| <b>DB2</b>                | DriverClass         | com.ibm.db2.jcc.DB2Driver                                                                                               |
|                           | Required .jar files | db2jcc.jar, db2cc_license_cu.jar                                                                                        |
|                           | URL                 | jdbc:db2://<hostname>:<dbport>/<dbname><br><br><b>Ex.</b><br>jdbc:db2://10.120.16.30:50001/WL814CS                      |
| <b>Oracle Thin driver</b> | DriverClass         | oracle.jdbc.driver.OracleDriver                                                                                         |
|                           | Required .jar files | ojdbc14.jar                                                                                                             |
|                           | URL                 | jdbc:oracle:thin:@//<hostname>:1521/<dbname><br><br><b>Ex.</b><br>jdbc:oracle:thin:@//godzilla.fatwire.com:1521/LINKSYS |

**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.)

9. Create a new file named `server.xml`, paste in the XML code below, and make the following changes:
  - a. If you are using Tomcat 6, replace the following line:
 

```
"<Listener
  className="org.apache.catalina.storeconfig.StoreConfigLifecycleListener" />"
```

 with:
 

```
"<Listener
  className="org.apache.catalina.core.JasperListener" />"
```
  - b. Set the `driverClassName` and `url` to the values you selected from [Table 1](#) in [step 8](#).
  - c. Set the `username` and `password`.

```
<Server port="8005" shutdown="SHUTDOWN">

  <Listener
className="org.apache.catalina.core.AprLifecycleListener" />
  <Listener
className="org.apache.catalina.mbeans.ServerLifecycleListener" />
  <Listener
className="org.apache.catalina.mbeans.GlobalResourcesLifecycleList
ener" />
  <Listener
className="org.apache.catalina.storeconfig.StoreConfigLifecycleLis
tener"/>

  <GlobalNamingResources>
    <Resource name="UserDatabase" auth="Container"
      type="org.apache.catalina.UserDatabase"
      description="User database that can be updated and saved"

factory="org.apache.catalina.users.MemoryUserDatabaseFactory"
      pathname="conf/tomcat-users.xml" />

  </GlobalNamingResources>

  <Service name="Tomcat-standalone">
    <Connector port="8080"
      maxThreads="150" minSpareThreads="25"
maxSpareThreads="75"
      enableLookups="false" redirectPort="8443"
acceptCount="100"
      connectionTimeout="20000"
      disableUploadTimeout="true" />

    <!-- SSL -->
    <!--
    <Connector port="8443" maxHttpHeaderSize="8192"
      maxThreads="150" minSpareThreads="25"
maxSpareThreads="75"
      enableLookups="false" disableUploadTimeout="true"
      acceptCount="100" scheme="https" secure="true"
      clientAuth="false" sslProtocol="TLS" />
    -->

    <Connector port="8009"
      enableLookups="false" redirectPort="8443"
      protocol="AJP/1.3" />

  <Engine name="Standalone" defaultHost="localhost">
    <Realm
className="org.apache.catalina.realm.UserDatabaseRealm"
      resourceName="UserDatabase"/>
    <Host name="localhost" appBase="webapps"
      unpackWARs="true"
```

```
    autoDeploy="true"
    xmlValidation="false"
    xmlNamespaceAware="false">
      <Context path="/cs" docBase="cs" reloadable="true"
crossContext="true">
        <Resource name="csDataSource" auth="Container"
type="javax.sql.DataSource"
            maxActive="100" maxIdle="30"
            username="insert-user"
            password="insert-password"
            driverClassName="insert-driver-class"
            url="insert-connection-url"/>
      </Context>
<!-- Clustering -->
<!--
  <Cluster
className="org.apache.catalina.cluster.tcp.SimpleTcpCluster"
        doClusterLog="true"
        clusterLogName="clusterlog"

managerClassName="org.apache.catalina.cluster.session.DeltaManager
"
        expireSessionsOnShutdown="false"
        useDirtyFlag="true"
        notifyListenersOnReplication="true">

  <Membership

className="org.apache.catalina.cluster.mcast.McastService"
        mcastAddr="228.0.0.4"
        mcastPort="45564"
        mcastFrequency="500"
        mcastDropTime="3000"/>

  <Receiver

className="org.apache.catalina.cluster.tcp.ReplicationListener"
        tcpListenAddress="auto"
        tcpListenPort="4001"
        tcpSelectorTimeout="100"
        tcpThreadCount="6"/>

  <Sender

className="org.apache.catalina.cluster.tcp.ReplicationTransmitter"
        replicationMode="pooled"
        ackTimeout="15000"/>

  <Valve

className="org.apache.catalina.cluster.tcp.ReplicationValve"

filter=".*\.gif;.*\.js;.*\.jpg;.*\.htm;.*\\.html;.*\\.txt;"/>
```

```

        <Deployer
className="org.apache.catalina.cluster.deploy.FarmWarDeployer"
        tempDir="/tmp/war-temp/"
        deployDir="/tmp/war-deploy/"
        watchDir="/tmp/war-listen/"
        watchEnabled="false"/>

        <ClusterListener
className="org.apache.catalina.cluster.session.ClusterSessionListe
ner"/>
    </Cluster>
    -->
    </Host>
</Engine>

</Service>
</Server>

```

10. Edit the `setclasspath.sh` file as follows:

a. Locate the following section:

```

# Set standard CLASSPATH
if [ "$1" = "debug" -o "$1" = "javac" ] ; then
    CLASSPATH="$JAVA_HOME/lib/tools.jar
fi

```

b. Comment out the if-statement lines marked in bold. The modified section should look as follows:

```

# Set standard CLASSPATH
#if [ "$1" = "debug" -o "$1" = "javac" ] ; then
    CLASSPATH="$JAVA_HOME/lib/tools.jar
#fi

```

c. Save and close the file.

11. In this step, you will test the new instance by starting it and displaying the test image. Do the following:

a. Make sure that the main instance (`$CATALINA_HOME`) was shut down before trying to start the new instance.

Because the two instances use the same ports, they will not start simultaneously. For the main instance to be shut down, `$CATALINA_BASE` must be empty before you run the shutdown command.

b. To start the new instance, make sure `$CATALINA_BASE` is set to the new instance `cs_tomcat` directory (or the directory name that you provided). Run the following command:

```
$CATALINA_HOME/bin/startup.sh
```

c. In your browser, open `http://<hostname>:8080/cs/tomcat.gif`

If the `tomcat.gif` image is displayed, the setup was done properly.

12. For the next step, see the “[Installation Quick Reference](#),” on page 7.

## Setting Up Multiple Instances of Tomcat to Run Simultaneously

This section explains the extra steps that need to be taken if multiple Tomcat instances (including the main `$CATALINA_HOME` instance) must be run simultaneously—**regardless of the applications they are serving**. The steps ensure that no two instances use the same ports:

1. Shut down an instance, as it must be restarted whenever changes to the `server.xml` are made:

```
$CATALINA_HOME/bin/shutdown.sh -force
```

2. Check that the `server.xml` file uses unique port values for the Tomcat instances. The port values to check are:
  - a. The `port` value in the first `connector` tag (set to 8080 in the main instance and previous example). This is the main port where the application will be deployed.
  - b. The `shutdown port` value, located in the `Server` tag (with a value of 8005 in the example above as well as the main instance). If this value is identical for two instances attempting to run simultaneously, the second instance will be unable to start.
  - c. The `redirectPort` value in both `Connector` tags. The value in the sample code in [step 9 on page 21](#) above is 8443 (the default value).
  - d. The `connector port` value, located in the second `Connector` tag (with a value of 8009 in the example in [step 9 on page 21](#)).
3. Test whether two instances can be run simultaneously:
  - a. Make sure both instances have been shut down, then set `$CATALINA_BASE` to the first instance, and run the `startup` command:

```
export CATALINA_BASE=<path to 1st cs_tomcat dir>
$CATALINA_HOME/bin/startup.sh
```
  - b. Set `$CATALINA_BASE` to the second instance, and run the `startup` command:

```
export CATALINA_BASE=<path to 2nd cs_tomcat dir>
$CATALINA_HOME/bin/startup.sh
```
  - c. To verify that the two instances can be run simultaneously, display the image `tomcat.gif`:

```
http://<hostname_for_2nd_instance>:<port_for_2nd_instance>/
cs/tomcat.gif
```

If there is an error, check both `server.xml` files to make sure the instances are not sharing any ports.
4. For the next step, see the “[Installation Quick Reference](#),” on [page 7](#).

## Configuring Tomcat for SSL

### Note

This section applies only to single-server CS installations.

1. Generate a certificate:

```
keytool -genkey -alias tomcat -keyalg RSA -keystore
$CATALINA_BASE/keys/.keystore
```

  - a. Set the `-keystore` parameter to the location where you want the generated keys to be stored.
  - b. When prompted for the keystore password, you may use your own password, or the default “changeit”. If a custom password is entered, a value in the `server.xml` file will need to be changed as explained later.
  - c. Enter your first and last name, name of organization, city, state, and country code when prompted, then click **Yes** to confirm.
  - d. At the next prompt, “Enter key password for <tomcat> (RETURN if same as keystore password),” press Enter, as the key password **must** be the same as the keystore password.
2. Edit `$CATALINA_HOME/conf/server.xml`, uncomment the SSL section, and add the following attributes to the SSL connector:
  - `keystoreType="JKS"`  
This attribute is set to JKS for “Java Keystore” (the format produced by Java’s `keytool`).
  - `keystoreFile="<path to $CATALINA_BASE>/keys/.keystore"`  
This attribute is set to the path where the `.keystore` file was created and the same path used in [step 1](#).
  - `keystorePass="<new password>"`  
This attribute is needed only if the keystore password used in [step 1](#) is not “changeit”, and should be set to the custom password used.
  - a. If the `redirectPort` values in the `<Connector>` tags are not already set to 8443, change them now.
3. Point your browser to `http://<hostname>:8443/cs/tomcat.gif`.  
If your system works correctly, you will be prompted to accept the certificate. When you accept the certificate, the Tomcat image is displayed.
4. For the next step, see the “[Installation Quick Reference](#),” on [page 7](#).

## Part 3

# Web Server

This part shows you how to install and configure Apache Web Server.

This part contains the following chapter:

- [Chapter 4, “Integrating with Apache Web Server”](#)



## Chapter 4

# Integrating with Apache Web Server

The chapter shows you how to install and integrate Apache Web Server with Tomcat Application Server. The steps in this chapter are required if you wish to integrate Tomcat with Apache Web Server or to create a vertical cluster that will be load balanced through Apache Web Server.

This chapter contains the following sections:

- [Step I. Install Apache Web Server 2.0.x](#)
- [Step II. Integrate Tomcat with Apache Web Server](#)

## Step I. Install Apache Web Server 2.0.x

1. Make sure the Tomcat instance that you will be using with Apache runs on the default ports, as only these ports are recognized.
2. Install Apache 2.0.x.
3. Set the variable `$APACHE2_HOME` to the directory in which Apache was installed.
4. Download and compile the newest release of `mod_jk`:
  - a. After downloading `mod_jk`, untar it using the following commands:

```
gunzip jakarta-tomcat-connectors-1.2.15-src.tar.gz
tar -xvf jakarta-tomcat-connectors-1.2.15-src.tar
```
  - b. Go to the `jakarta-tomcat-connectors-<version>-src/jk/native` directory:

```
cd jakarata-tomcat-connectors-1.2.15-src/jk/native
```
  - c. Configure and compile the `mod_jk.so` file:
    - 1) `./configure --with-apxs=$APACHE2_HOME/bin/apxs`
    - 2) `make`
    - 3) `cd apache-2.0`
    - 4) `$APACHE2_HOME/bin/apxs -n jk -i mod_jk.so`The last command automatically places the `mod_jk.so` file into your `$APACHE2_HOME/modules` directory.

## Step II. Integrate Tomcat with Apache Web Server

In this step you will use `mod_jk` to integrate Tomcat Application Server with Apache Web Server.

1. Create `workers.properties` in `$APACHE2_HOME/conf` with the following contents:

```
ps=/
worker.list=tomcat
worker.tomcat.port=<ajp port>
worker.tomcat.host=<hostname>
worker.tomcat.type=ajp13
worker.tomcat.lbfactor=1
```

The `<ajp port>` value can be found in the `server.xml` file in the Connector tag referencing the AJP protocol. The default value is 8009.

2. Edit `$APACHE2_HOME/conf/httpd.conf`:
  - a. Add the following to the `LoadModules` section:

```
LoadModule jk_module modules/mod_jk2.so
```
  - b. Add the following before Section 3: Virtual Hosts:

```
#
# mod_jk settings
#
```

```
JkWorkersFile "conf/workers.properties"  
JkLogFile "logs/mod_jk.log"  
JkLogLevel info  
JkMount /cs/* tomcat  
# End of mod_jk settings
```

3. Test httpd.conf by typing the following:

```
cd $APACHE2_HOME/bin  
apachectl configtest
```

If successful, you will receive a warning message and then "Syntax OK" . Ignore the warning.

4. Start Tomcat:

```
$CATALINA_HOME/bin/startup.sh
```

5. Start Apache:

```
$APACHE2_HOME/bin/apachectl start
```

6. Point your browser to `http://<hostname>/cs/tomcat.gif` and verify that the Tomcat image is displayed.



## Part 4

# Content Server

This part shows you how to install Content Server. It contains the following chapter:

- [Chapter 5, “Installing and Configuring Content Server”](#)



## Chapter 5

# Installing and Configuring Content Server

This chapter guides you through the installation of Content Server on the Tomcat Application Server.

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 completed [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. If you are using an Oracle database and require text attributes greater than 2000 characters, you must set the `cc.bigtext` property to `CLOB`. When the installer displays the “Installation Actions” pop-up window, complete step 1 displayed in the window, then do the following:
  - a. Open the Property Editor by clicking the **Property Editor** button.
  - b. In the Property Editor, open the `futuretense.ini` file.
  - c. Click the **Database** tab.

- d. Locate the `cc.bigtext` property and set its value to `CLOB`.
  - e. Save your changes and close the Property Editor.
  - f. Continue on to step 3 displayed in the “Installation Actions” window.
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 \(Unix Only\)](#)
- B. [Adding the CS Binaries Directory to the Library Path Variable](#)
- C. [Verifying the Installation](#)
- D. [Integrating with LDAP \(Optional\)](#)
- E. [Setting Up a Content Server Cluster \(Optional\)](#)
- F. [Setting Up Content Server for Its Business Purpose](#)

### 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 Tomcat.

### B. Adding the CS Binaries Directory to the Library Path Variable

In order for the components to function in Content Server, you must add the CS binaries directory, `<cs_install_dir>/bin`, to your library path variable as follows:

- For HP-UX:  
Add `<cs_install_dir>/bin` to `SHLIB_PATH`.
- For Linux and Solaris:  
Add `<cs_install_dir>/bin` to `LD_LIBRARY_PATH`.
- For AIX:  
Add `<cs_install_dir>/bin` to `LIBPATH`.
- For Windows:  
Add `<cs_install_dir>\bin` to your system’s `PATH` variable.

## C. Verifying the Installation

Verify the installation by logging in to Content Server as the administrator.

### 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.



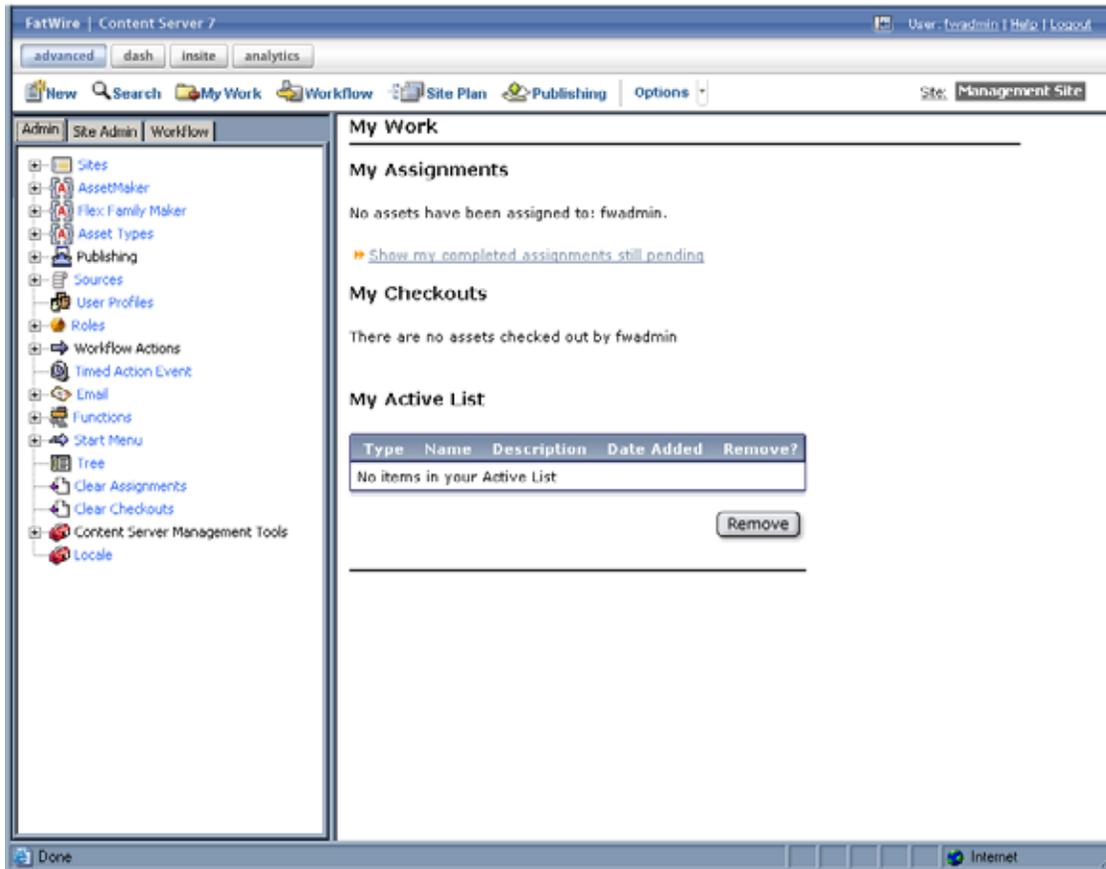
2. Enter the following credentials:

- User name: **fwadmin**
- Password: **xceladmin**

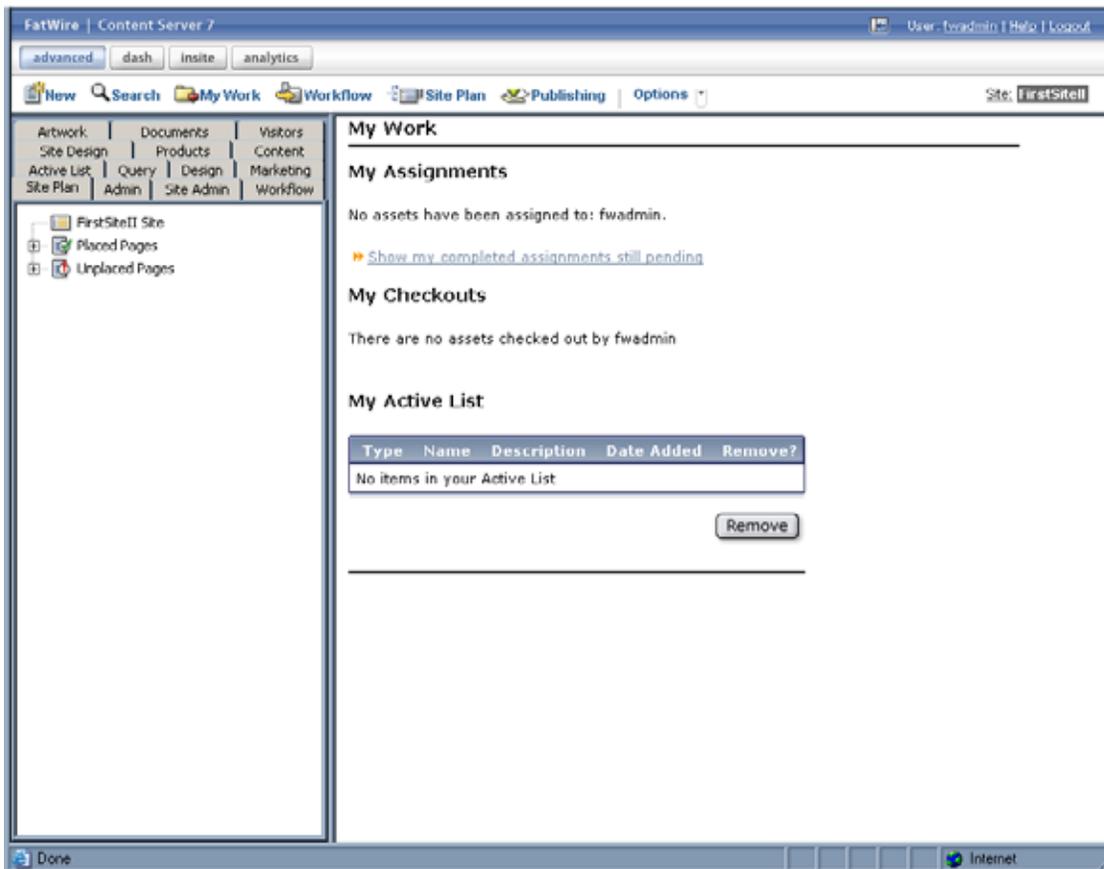
**3. Click Login.**

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.

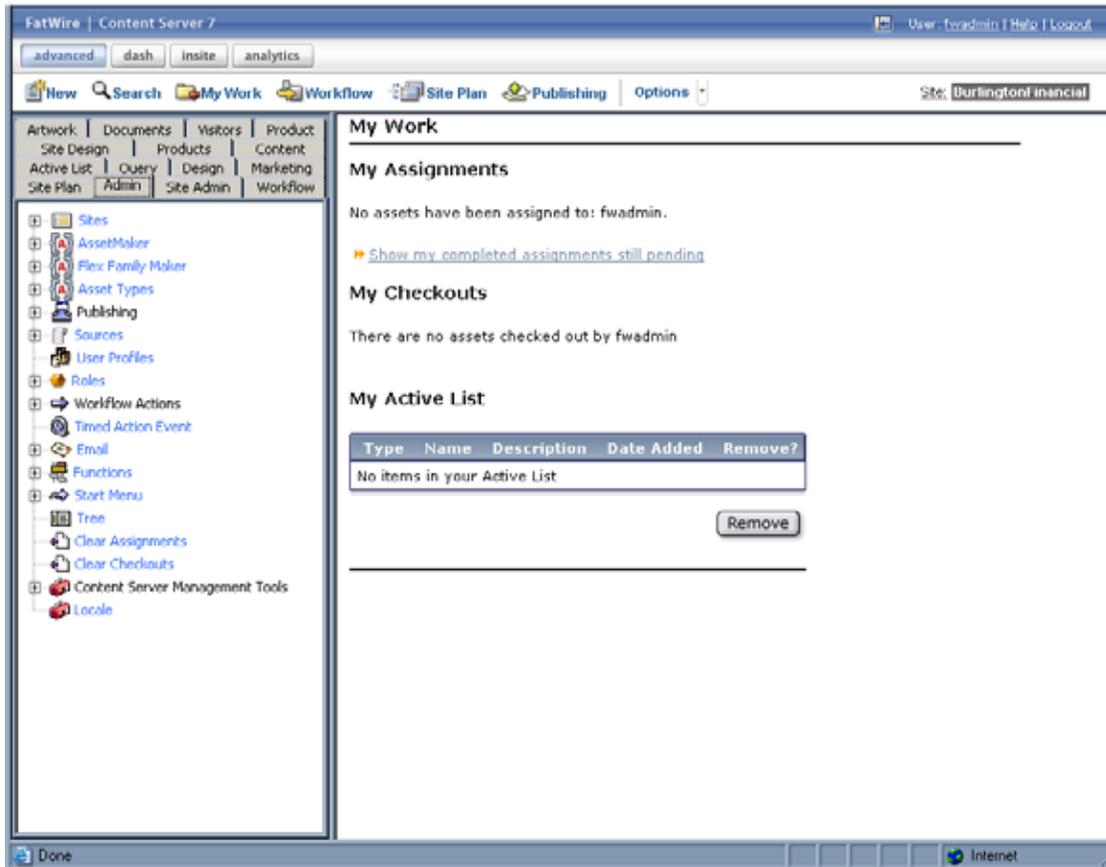
**You have logged in as fwadmin**

Select a site that you want to work on:

Site	Description	Assigned Role
<a href="#">BurlingtonFinancial</a>	Burlington Financial	GeneralAdmin, ArtworkEditor, Approver, ContentEditor, WorkflowAdmin, Analyst, Pricer, Marketer, SiteAdmin, Checker, MarketingAuthor, MarketingEditor, Author, Editor, ContentAuthor, Expert, ProductAuthor, ProductEditor, DocumentAuthor, DocumentEditor, Designer, ArtworkAuthor
<a href="#">FirstSiteII</a>	FirstSite Mark II	ArtworkEditor, GeneralAdmin, Approver, ContentEditor, WorkflowAdmin, Analyst, Pricer, Marketer, SiteAdmin, Checker, MarketingAuthor, MarketingEditor, Author, Editor, ContentAuthor, Expert, ProductAuthor, ProductEditor, DocumentAuthor, ArtworkAuthor, Designer, DocumentEditor
<a href="#">GE Lighting</a>	GE Lighting	Designer, SiteAdmin, WorkflowAdmin, GeneralAdmin

[\[Log in again\]](#)

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.

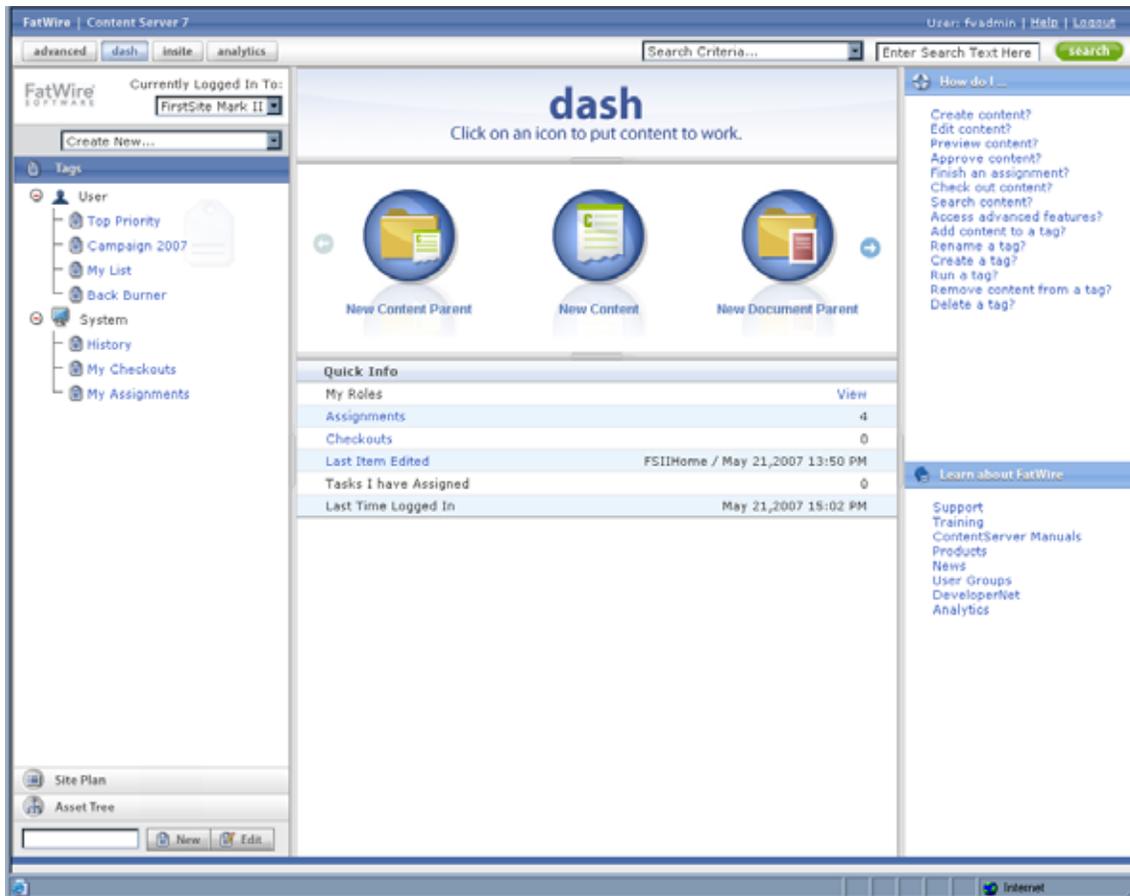


2. Enter the following credentials:
  - User name: **fwadmin**
  - Password: **xceladmin**

**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.



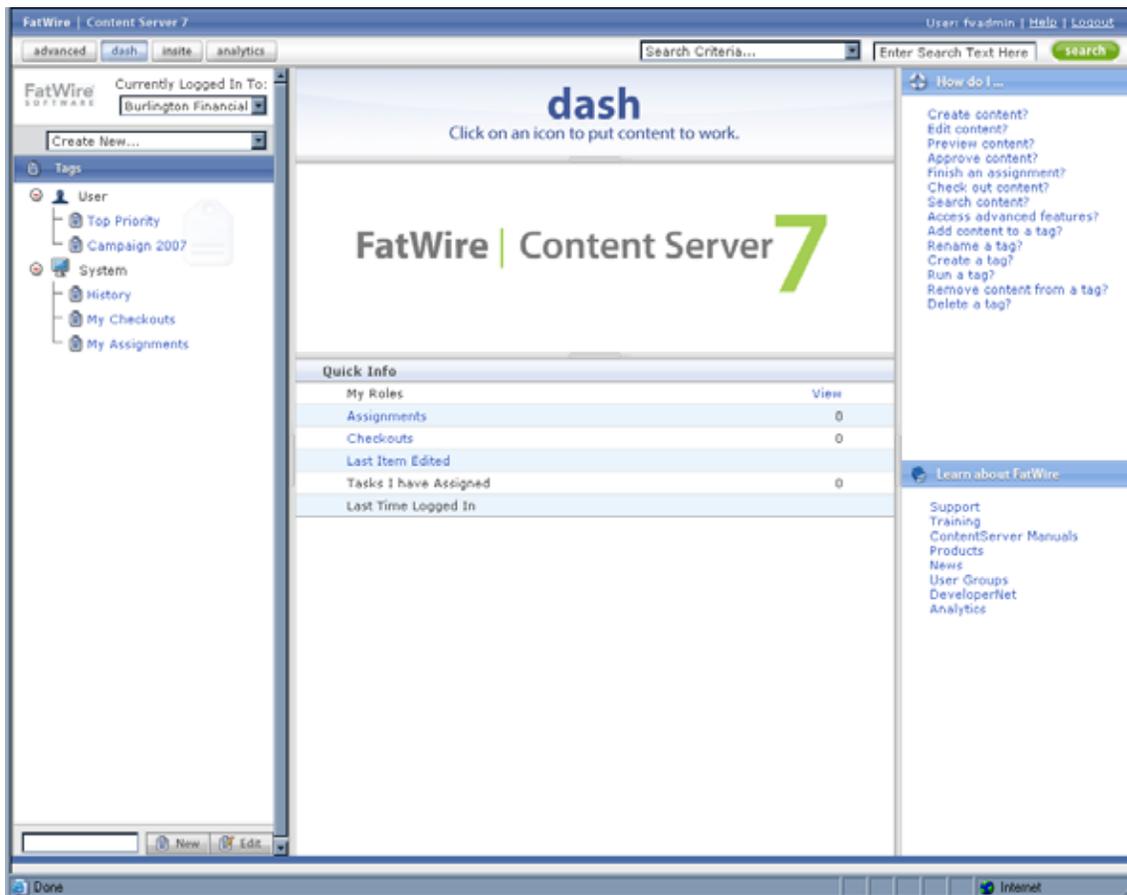
- If you installed more than one sample site, Content Server displays the “Select Site” screen. In such case, select the sample site you wish to log in to.

**You are currently logged in as 'fwadmin'**  
Select a site that you want to work on:

Select	Name	Description	Roles
<input type="radio"/>	BurlingtonFinancial	Burlington Financial	WorkflowAdmin, SiteAdmin, GeneralAdmin
<input type="radio"/>	GE Lighting	GE Lighting	Designer, WorkflowAdmin, SiteAdmin, GeneralAdmin
<input type="radio"/>	HelloAssetWorld	Hello Asset World	WorkflowAdmin, GeneralAdmin
<input type="radio"/>	FirstSiteII	FirstSite Mark II	GeneralAdmin

[ [log in again](#) ]

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.

## D. Integrating with LDAP (Optional)

If you wish 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*.

## E. Setting Up a Content Server Cluster (Optional)

This section is written on the assumption that you have already installed and configured Tomcat Application Server as specified in the previous sections of this guide. This means:

- You are installing a vertical cluster (Tomcat Application Server instances are installed on the same machine).
- You have installed and configured a single instance of Content Server to run through Tomcat. You have also logged in to the instance and confirmed that it is operational.

## 1. Creating Cluster Members

1. Create a new Tomcat instance and Content Server installation directory by following the steps in “[Setting Up Directories for Content Server and Tomcat](#),” on page 19.
2. Change the port settings of the new instance by following the steps in “[Setting Up Multiple Instances of Tomcat to Run Simultaneously](#),” on page 25.
3. Run the Content Server installer on the newly created 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 installation directory you created for the first instance of Content Server.
  - 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 to the primary cluster member. This defaults to the following:  
`<path_to_cs_install_dir_of_primaryClusterMember>/Shared.`
  - d. In the “Content Server URL Parameters” screen, enter the IP address and port you used in [step 2](#) of this procedure.
  - e. In the “Content Server Application Deployment” screen, enter the path to `CATALINA_BASE`.
4. For every member of the cluster, edit the `web.xml` file (located in `<path to cs_tomcat_directory>/webapps/cs/WEB-INF/`) by adding the line `<distributable/>` between the `<web-app>` and `<servlet>` line:

```
<web-app>
<distributable/>
<servlet>
```
5. For every member of the cluster, open the `server.xml` file and uncomment the Clustering section to set up Tomcat for clustering. Check the following:
  - a. All Tomcat instances belonging to the same cluster have the same values for `mcastAddr` and `mcastPort` in the `Membership` tag.
  - b. All Tomcat instances have a different value for `tcpListenPort` in the `Receiver` tag.
6. Make sure there is a `usedisksync` directory in the `Shared` directory of your primary CS installation, or the first one that was created. If the `usedisksync` directory is not already there, then create it with the following command:

```
mkdir <path_to_primary_cs_install_dir>/Shared/usedisksync
```
7. To finish configuring Content Server for clustering, edit the following properties by using Content Server’s Property Editor. **Complete the steps below for all cluster members:**
  - a. Launch the CS Property Editor (make sure your `DISPLAY` variable is set):

```
<cs_install_dir>/propeditor.sh
```
  - b. From the **File** menu, select **Open**, navigate to your Content Server installation directory and open the `futuretense.ini` file.
  - c. Click **Cluster**.

- d. Make sure the following variables are set as shown below:
  - `cc.cacheNoSync` is set to `false`
  - `ft.sync` is set to a value (assigned by you) that is identical for all members of this cluster, e.g., `cluster1`.
  - `ft.usedisksync` is set to the path of the `usedisksync` directory created in [step 6](#).
8. Verify the newly created cluster member. For instructions, see [“C. Verifying the Installation,”](#) on [page 38](#).
9. 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

For every member of the cluster, complete the following steps:

1. Open the `server.xml` file and add the following attribute to the Engine tag:

```
jvmRoute="node1"
```

### Note

Each member can be given any node name as long as the names are unique across members.

2. Make sure the `workers.properties` file 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
```

3. Use the names stored in the `server.xml` file mentioned in [step 1](#) of this section.

#### Note

[Step 2](#) supports two cluster members and one load balancer. For each additional cluster member, add the member name as well as the lines below to the `worker.list` and `worker.loadbalancer.balanced_workers` files:

```
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
```

4. Edit the `$APACHE2_HOME/conf/httpd.conf` file by adding the following line between the `<web-app>` and `<filter>` lines, as shown below:  

```
<web-app>
<istributable/>
<filter>
```
5. Load balancer configuration is now complete. Restart Apache for the changes to take effect.

## F. Setting Up Content Server for Its Business Purpose

You are now ready to configure Content Server 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, workflow operations, and client interfaces.