

# Content Server

---

Version: 6.3

## Installing Content Server on Tomcat Application Server

Document Revision Date: May 24, 2006



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 © 2006 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, Inc. 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 limitation 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 on Tomcat Application Server*

Document Revision Date: May 24, 2006

Product Version: 6.3

#### **FatWire Technical Support**

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

#### **FatWire Headquarters**

FatWire Corporation  
330 Old Country Road  
Suite 207  
Mineola, NY 11501  
[www.fatwire.com](http://www.fatwire.com)

Table of

## Contents

What This Guide Covers .....	6
What This Guide Does Not Cover.....	6
Installation Summary.....	6
System Requirements .....	6
Graphics in This Guide .....	7
Installation Steps .....	7

### Part 1. Database

<b>2 Setting Up a Database .....</b>	<b>11</b>
--------------------------------------	-----------

### Part 2. Application Server

<b>3 Installing and Configuring Tomcat.....</b>	<b>15</b>
Installing Tomcat .....	16
Configuring Tomcat .....	17
Setting Up Directories .....	17
Running Multiple Instances of Tomcat Simultaneously .....	25
Important Commands .....	26
Configuring Tomcat for SSL.....	26

### Part 3. Web Server

<b>4 Integrating Tomcat with Apache Web Server .....</b>	<b>31</b>
Step I. Install the Apache Web Server 2.0.x .....	32

Step II. Configure mod_jk for Apache .....	32
--	----

## **Part 4. Content Server**

<b>5 Installing and Deploying Content Server.....</b>	<b>37</b>
Step I. Complete Pre-Installation Procedures .....	38
Step II. Install Content Server .....	38
Step III. Complete Post-Installation Procedures .....	60
Installing Content Server in a Clustered Environment .....	60
Load Balancing with mod_jk.....	62

<b>Appendix A. Installing Verity Search Engine .....</b>	<b>65</b>
--	-----------

## Chapter 1

# Installation Overview

This document provides guidelines for installing Content Server on the Tomcat Application Server, connecting to the 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:

- [What This Guide Covers](#)
- [What This Guide Does Not Cover](#)
- [Installation Summary](#)
- [System Requirements](#)
- [Graphics in This Guide](#)
- [Installation Steps](#)

## What This Guide Covers

This guide covers the installation, configuration, and maintenance of the Tomcat Application Server, as required to support Content Server. This includes configuration of one or more Tomcat instances, backend databases, and integrating Tomcat with the Apache Web Server.

This guide also provides instructions for installing Content Server in both vertically clustered and non-clustered environments.

## What This Guide Does Not Cover

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

- Tomcat SSL configuration for simultaneously run instances
- IIS and Sun Web Server installation, as no plugins exist that support these web servers
- LDAP integration

## Installation Summary

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 or bypass sample sites and sample content, depending on the system you are setting up and on your business needs.

### Note

The names of the systems in your Content Server environment might vary from the names used in this document. Generally, the management system is also called “staging,” and the delivery system is also called “production.”

## System Requirements

System requirements for installing Content Server are given in the following documents, located on your Content Server installation CD:

- *Content Server Supported Platform Document*, which specifies third-party databases and drivers, application servers and web servers, and other software required for installing and running Content Server.
- *Content Server Release Notes*, which provide important information about Content Server.

FatWire recommends that you read both of these documents before installing Content Server.

### Note

The latest versions of the above-mentioned documents are located at the following URL (password-protected):

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

If you need a password, contact FatWire Technical Support. Contact information is available at the following URL:

`http://www.fatwire.com/Support/contact_info.html`

The e-docs website is organized by product and version number. To obtain the correct documents, follow the link for the version of Content Server you are installing.

## 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. The 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 Steps

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

### To install Content Server and its supporting software

1. Ensure that you have licensed copies of all the software you will be installing. For information about Content Server's supporting software, refer to the *Content Server Supported Platform List* and to *Release Notes*. The latest versions are available on the e-docs website (password-protected), at the URL that is given in "[System Requirements](#)," on page 6.
2. Set up your choice of supported databases by installing the database management system, creating a database for Content Server, and configuring the database. For instructions, see our configuration guide, *Third-Party Software*.
3. Install and configure Tomcat Application Server. For instructions, see [Chapter 3](#), "[Installing and Configuring Tomcat](#)."

### Note

In addition to basic installation/configuration steps, [Chapter 3](#) provides instructions for running multiple instances of Tomcat simultaneously and using SSL.

4. If you plan to integrate Tomcat with the Apache web server, follow instructions in [Chapter 4, “Integrating Tomcat with Apache Web Server.”](#)
5. Install Content Server by running the installer (halfway through the installation, you will deploy Content Server). For instructions, see [Chapter 5, “Installing and Deploying Content Server.”](#)
6. If you plan to use the Verity search engine, follow installation guidelines in [Appendix A, “Installing Verity Search Engine.”](#)

## 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 our configuration guide, *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 supported database that is specifically configured for Content Server.

The complete 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>).

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 configuration guide, *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 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:

- [Installing Tomcat](#)
- [Configuring Tomcat](#)

## 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 the Java Development Kit from Sun:
  - For Tomcat 5.0, use version 1.4.x of the JDK.
  - For Tomcat 5.5, use version 1.5.x of the JDK.
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:
  - For Tomcat 5.0:

```
gzip -d jakarta-tomcat-5.0.28.tar.gz
tar -xvf jakarta-tomcat-5.0.28.tar
```
  - For Tomcat 5.5:

```
gzip -d apache-tomcat-5.5.16.tar.gz
tar -xvf apache-tomcat-5.5.16.tar
```
5. Set the `JAVA_HOME` variable to the JDK folder of the version of Java that will be used. For example:

```
export JAVA_HOME=/u01/software/App/jdk142_05
```
6. Start the application server by running the startup command.

```
$CATALINA_HOME/common/bin/startup.sh
```
7. Access the following URL in a web browser: `http://<hostname>:8080/`  
This brings you to Tomcat's default homepage.
8. Shut down the application server by running the `shutdown` command.

```
$CATALINA_HOME/common/bin/shutdown.sh
```
9. Edit the `$CATALINA_HOME/conf/tomcat-users.xml` file and add the following lines:

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

This creates a new user with the permissions to access both the Admin and Manager tools.
10. You have successfully installed Tomcat. Proceed to the next section, "[Configuring Tomcat](#)," on page 17.

## Configuring Tomcat

This section contains the following information and instructions:

- [Setting Up Directories](#)
- [Running Multiple Instances of Tomcat Simultaneously](#)
- [Important Commands](#)
- [Configuring Tomcat for SSL](#)

### Setting Up Directories

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

```
mkdir CS6.3
cd CS6.3
```

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:

```
CS6.3/
  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.

Microsoft SQL Server	MS SQL
DriverClass	<code>com.microsoft.jdbc.sqlserver.SQLServerDriver</code>
Required .jar files	<code>mssqlserver.jar</code> <code>msbase.jar</code> <code>msutil.jar</code>
URL	<code>jdbc:microsoft:sqlserver://&lt;hostname&gt;:1433;SelectMethod=Cursor;DatabaseName=&lt;dbname&gt;</code>  E.g., <code>jdbc:microsoft:sqlserver://10.120.14.22:1433;SelectMethod=Cursor;DatabaseName=CS62_tomcat</code>
JTDS	Third-party driver
DriverClass	<code>net.sourceforge.jtds.jdbc.Driver</code>
Required .jar files	<code>jtds-1.1.jar</code>
URL	<code>jdbc:jtds:sqlserver://&lt;hostname&gt;:1433/&lt;dbname&gt;</code>  E.g., <code>jdbc:jtds:sqlserver://10.120.14.22:1433/CS62_tomcat</code>
DB2	
DriverClass	<code>com.ibm.db2.jcc.DB2Driver</code>
Required .jar files	<code>db2jcc.jar</code> , <code>db2cc_license_cu.jar</code>
URL	<code>jdbc:db2://&lt;hostname&gt;:&lt;dbport&gt;/&lt;dbname&gt;</code>  E.g., <code>jdbc:db2://10.120.16.30:50001/WL814CS</code>

Oracle	Thin driver
DriverClass	oracle.jdbc.driver.OracleDriver
Required .jar files	ojdbc14.jar, orai18n.jar
URL	jdbc:oracle:thin:@//<hostname>:1521/<dbname> E.g., jdbc:oracle:thin:@/ godzilla.fatwire.com:1521/ LINKSYS

9. Create a new file named `server.xml` and paste in the XML code below. Replace the `<Url>` and `<Driver Class>` text fragments with the URL and driver class selected from [step 8](#).

**For Tomcat 5.0:**

```
<Server port="8005" shutdown="SHUTDOWN" debug="0">

  <ListenerclassName="org.apache.catalina.mbeans.ServerLifecycleL
    istener" debug="0"/>

  <ListenerclassName="org.apache.catalina.mbeans.GlobalResourcesL
    ifecycleListener" debug="0"/>

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

          <value>org.apache.catalina.users.MemoryUserDatabaseFactory</value>
        </parameter>
        <parameter>
          <name>pathname</name>
          <value>conf/tomcat-users.xml</value>
        </parameter>
      </ResourceParams>
    </GlobalNamingResources>

    <Service name="Tomcat-standalone">
      <Connector port="8080"
        maxThreads="150" minSpareThreads="25"
        maxSpareThreads="75"
        enableLookups="false" redirectPort="8443"
        acceptCount="100"
        debug="0" connectionTimeout="20000"
        disableUploadTimeout="true" />
```

```

    <!-- SSL -->
    <!--
    <Connector port="8443"
        maxThreads="150" minSpareThreads="25"
maxSpareThreads="75"
        enableLookups="false" disableUploadTimeout="true"
        acceptCount="100" debug="0" scheme="https"
secure="true"
        clientAuth="false" sslProtocol="TLS" />
    -->

    <Connector port="8009"
        enableLookups="false" redirectPort="8443" debug="0"
        protocol="AJP/1.3" />

    <Engine defaultHost="localhost" name="Standalone" debug="0">
        <Logger className="org.apache.catalina.logger.FileLogger"
            prefix="localhost_log." suffix=".txt" timestamp="true"/>
        <Realm
            className="org.apache.catalina.realm.UserDatabaseRealm" debug="0"
            resourceName="UserDatabase"/>
        <Host appBase="webapps" name="localhost" debug="0"
            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" />
                <ResourceParams name="csDataSource">
                    <parameter><name>username</name><value>insert-
user</value></parameter>
                    <parameter><name>password</name><value>insert-
password</value></parameter>
                    <parameter><name>driverClassName</
name><value>insert-driver-class</value></parameter>
                    <parameter><name>url</name><value>insert-
connection-url</value></parameter>
                    <parameter><name>maxActive</name><value>100</
value></parameter>
                    <parameter><name>maxIdle</name><value>30</value></
parameter>
                </ResourceParams>
            </Context>

            <!-- Clustering -->
            <!--
            <Cluster
                className="org.apache.catalina.cluster.tcp.SimpleTcpCluster"

managerClassName="org.apache.catalina.cluster.session.DeltaManager"
"
                expireSessionsOnShutdown="false"

```

```
        useDirtyFlag="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"/>

        <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"/>
    </Cluster>
    -->

    </Host>
</Engine>

</Service>
</Server>
```

**For Tomcat 5.5:**

```

<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.** In this step, you will test the new instance by starting it and trying to display 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.

**11.** If you plan on running multiple instances of Tomcat, or using SSL, or integrating Tomcat with Apache web server, continue reading instructions in this step. Otherwise, go to [step 12](#).

- If you plan on running the new instance of Tomcat at the same time as the main instance of Tomcat (`$CATALINA_HOME`), or any other instance configured with these port numbers, additional changes must be made to the `server.xml` file as explained in the next section. Complete the steps in [“Running Multiple Instances of Tomcat Simultaneously.”](#)
- If you plan to use SSL, you need to configure Tomcat for SSL. Complete the steps in [“Configuring Tomcat for SSL,”](#) on page 26.
- If you wish to integrate Tomcat with Apache web server, go to [Chapter 4](#), [“Integrating Tomcat with Apache Web Server](#) for instructions.

12. Configuration of the Tomcat Application Server is complete. You are now ready to install Content Server. Go to [Chapter 5, “Installing and Deploying Content Server”](#) for instructions.

## Running Multiple Instances of Tomcat Simultaneously

This section explains the extra steps that need to be taken if multiple Tomcat instances (including the main `$CATALINA_HOME` instance) will be running simultaneously.

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

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

2. Multiple instances must have the following values differ in their `server.xml` file:
  - a. The `port` value in the first connector tag (set to 8080 in the main instance and previous example) must be different for all other instances being run at the same time. This is the main port where the application will be deployed.
  - b. The `shutdown port` value is located in the Server tag, with a value of 8005 in the example above as well as the main instance. If this value is not different for two instances attempting to be run simultaneously, the second instance will be unable to start.
  - c. In both Connector tags, the `redirectPort` value must be different between instances. The value in the example above is 8443 and is also the default value.
  - d. The last change that needs to be made is to the `connector port` value, which is located in the second Connector tag, with a value of 8009 in the example in [step 9 on page 19](#). Make sure this value is also different between instances.
3. Test whether two instances can be run simultaneously:
  - a. Make sure both instances have been shut down, then set `$CATALINA_BASE` to one, and run the `startup` command.
  - b. Set `$CATALINA_BASE` to the second, and run the `startup` command.

```
export CATALINA_BASE=<path to cs_tomcat1 or other tomcat dir>
$CATALINA_HOME/bin/startup.sh
export CATALINA_BASE=<path to cs_tomcat2>
$CATALINA_HOME/bin/startup.sh
```

If the `cs` instance was set up correctly for running simultaneously with the other instance, you should be able to run `http://<hostname for 2nd instance>:<port for 2nd instance>/cs/tomcat.gif` in your browser and have the image be displayed. If there is an error, check both `server.xml` files to make sure the instances are not sharing any ports.

4. If you plan on using SSL or integrating Tomcat with Apache web server, continue reading instructions in this step. Otherwise, go to [step 5](#).
  - If you plan to use SSL, you need to configure Tomcat for SSL. Complete the steps in [“Configuring Tomcat for SSL,” on page 26](#).
  - If you wish to integrate Tomcat with Apache web server, go to [Chapter 4, “Integrating Tomcat with Apache Web Server”](#) for instructions.

5. Configuration of Tomcat Application Server is complete. You are now ready to install Content Server. Go to [Chapter 5, “Installing and Deploying Content Server”](#) for instructions.

## Important 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</code>

## Configuring Tomcat for SSL

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, country code when prompted, and 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:
  - a. `keystoreType="JKS"`  
The “`keystoreType`” attribute is set to JKS for “Java Keystore” (the format produced by Java’s `keytool`).
  - b. `keystoreFile="<path to $CATALINA_BASE>/keys/.keystore"`  
The “`keystoreFile`” attribute is set to the path where the `.keystore` file was created and the same path used in [step 1](#).
  - c. `keystorePass="<new password>"`

The “`keystorePass`” attribute is needed only if the keystore password used in [step 1](#) is something other than “`changeit`”, and should be set to the custom password used.

- d. 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. If you plan on integrating Tomcat with Apache web server, go to “[Chapter 4, “Integrating Tomcat with Apache Web Server”](#)” for instructions. Otherwise, the configuration of Tomcat Application Server is complete and you are ready to install Content Server. Go to [Chapter 5, “Installing and Deploying Content Server”](#) for instructions.



## Part 3

# Web Server

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

This part contains the following chapter:

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



## Chapter 4

# Integrating Tomcat with Apache Web Server

The chapter shows you how to install and configure the Apache Web Server for integration with Tomcat Application Server.

This chapter contains the following sections:

- [Step I. Install the Apache Web Server 2.0.x](#)
- [Step II. Configure mod\\_jk for Apache](#)

## Step I. Install the 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`.
5. 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
```

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

```
cd jakarata-tomcat-connectors-1.2.15-src/jk/native
```

7. 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 automatically places the `mod_jk.so` file into your `$APACHE2_HOME/modules` directory.

## Step II. Configure `mod_jk` for Apache

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

```
ps=/
worker.list=tomcat
worker.jboss.port=<ajp port>
worker.jboss.host=<hostname>
worker.jboss.type=ajp13
worker.jboss.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.
7. You are now ready to install Content Server. Go to [Chapter 5, "Installing and Deploying Content Server"](#) for instructions.



## Part 4

# Content Server

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

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



## Chapter 5

# Installing and Deploying Content Server

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

This chapter contains the following sections:

- [Step I. Complete Pre-Installation Procedures](#)
- [Step II. Install Content Server](#)
- [Step III. Complete Post-Installation Procedures](#)
- [Installing Content Server in a Clustered Environment](#)

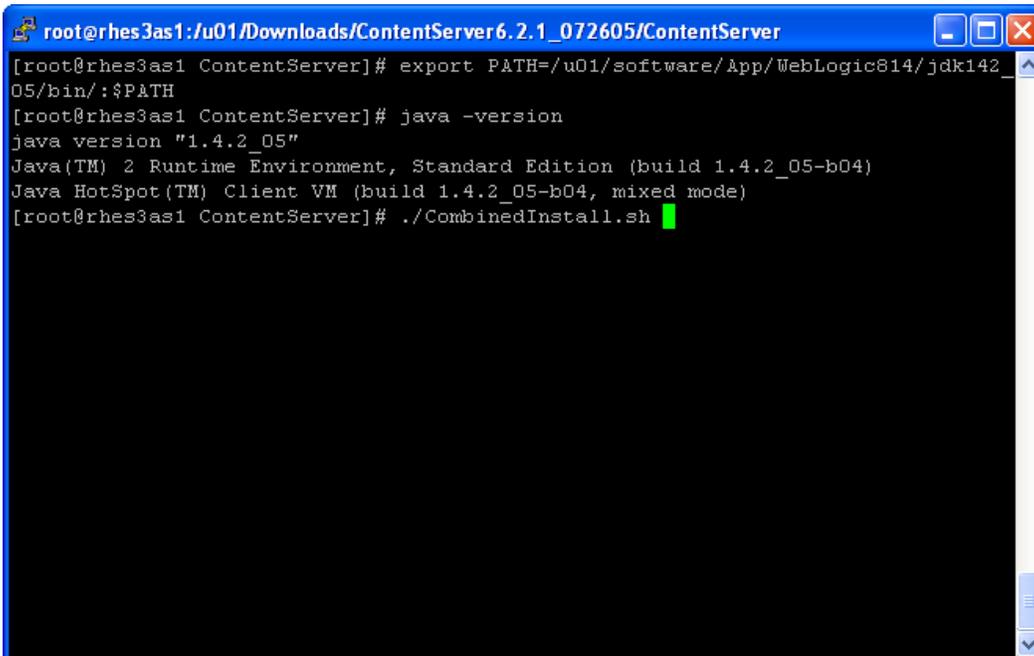
## Step I. Complete Pre-Installation Procedures

Before installing Content Server, make sure you have completed the following:

- Performed [steps 1–3](#) in the section “[Installation Steps](#),” on [page 7](#).
- Obtained a license with the correct IP address and ports
- Created a valid directory into which you plan to install Content Server
- For clustered installations: You have created a valid shared directory into which you plan to install the Content Server shared file system.

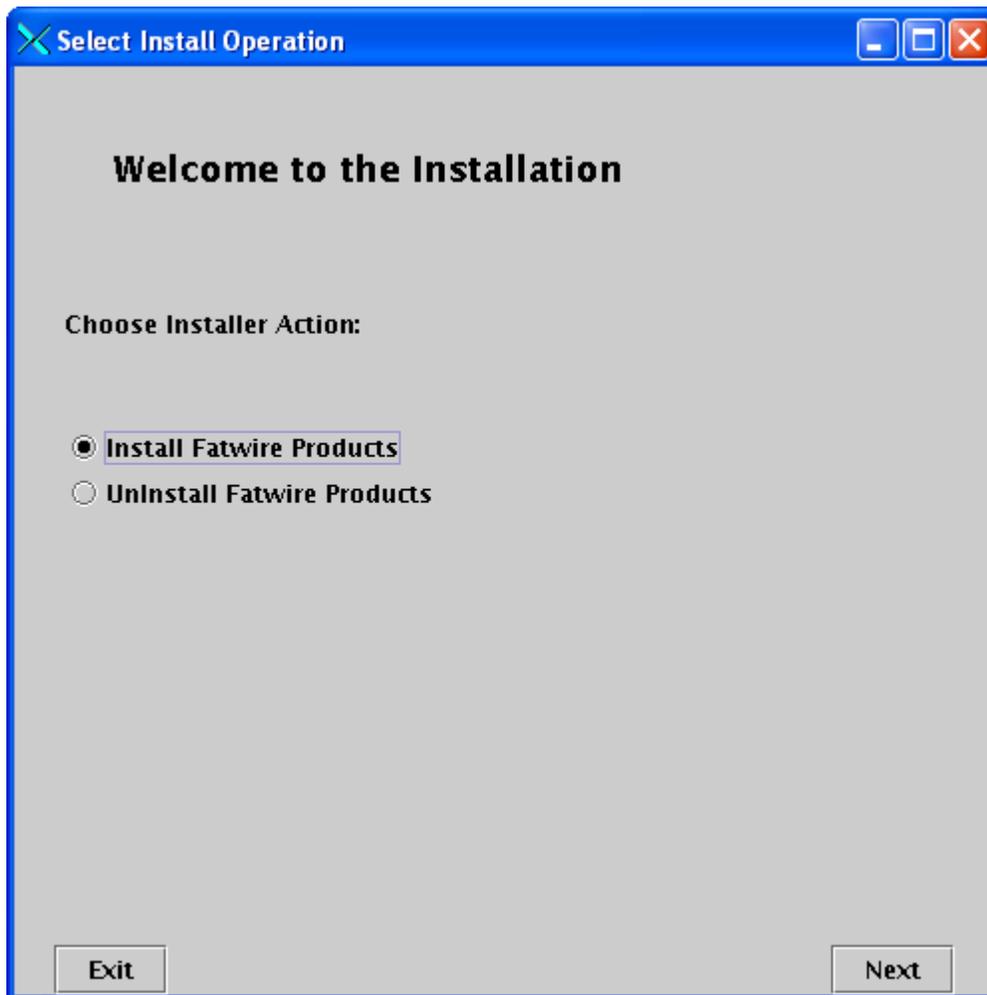
## Step II. Install Content Server

1. Start the Content Server installer by changing to the Content Server installer directory and running `./CombinedInstall.sh`

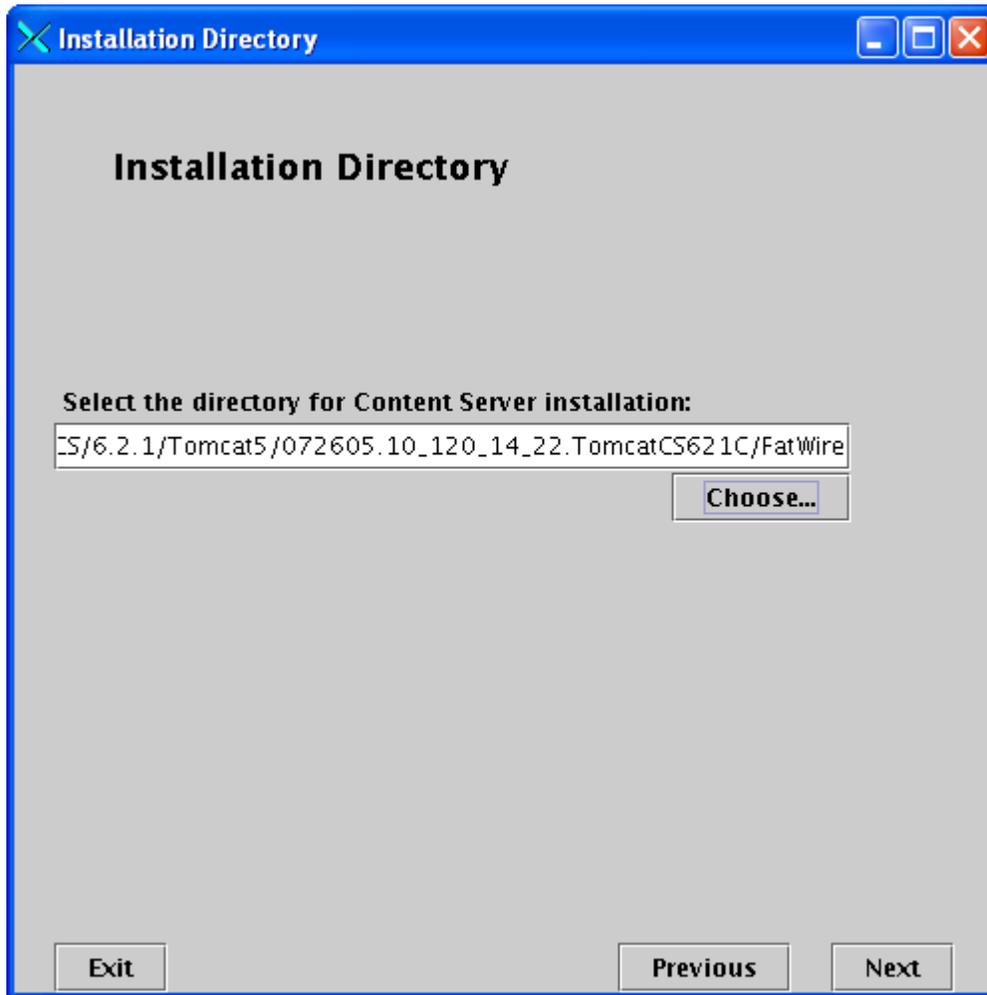
A terminal window titled "root@rhes3as1:/u01/Downloads/ContentServer6.2.1\_072605/ContentServer" is shown. The terminal output displays the following commands and their results:

```
[root@rhes3as1 ContentServer]# export PATH=/u01/software/App/WebLogic814/jdk142_05/bin:$PATH
[root@rhes3as1 ContentServer]# java -version
java version "1.4.2_05"
Java(TM) 2 Runtime Environment, Standard Edition (build 1.4.2_05-b04)
Java HotSpot(TM) Client VM (build 1.4.2_05-b04, mixed mode)
[root@rhes3as1 ContentServer]# ./CombinedInstall.sh
```

2. Select the option **Install FatWire Products** and click **Next**.



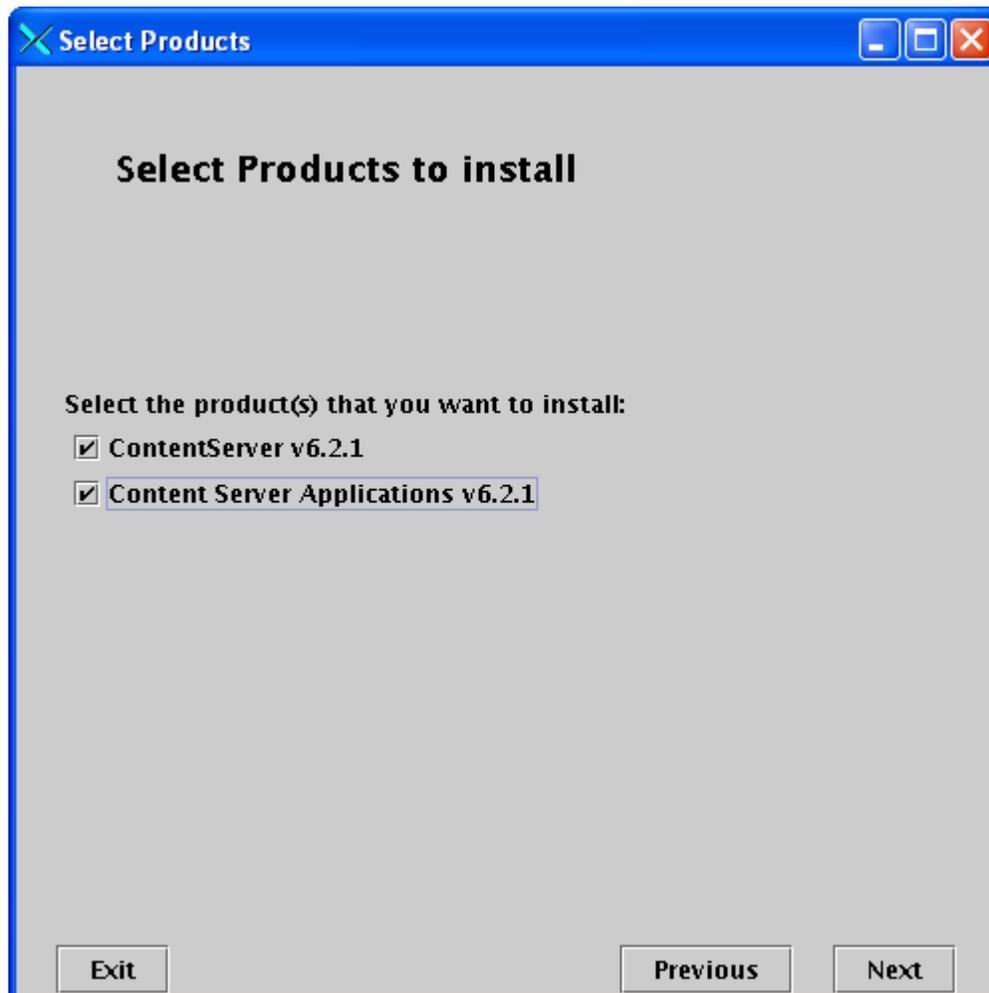
3. Choose the Content Server installation directory (the directory that was created in the pre-installation phase) and click **Next**.



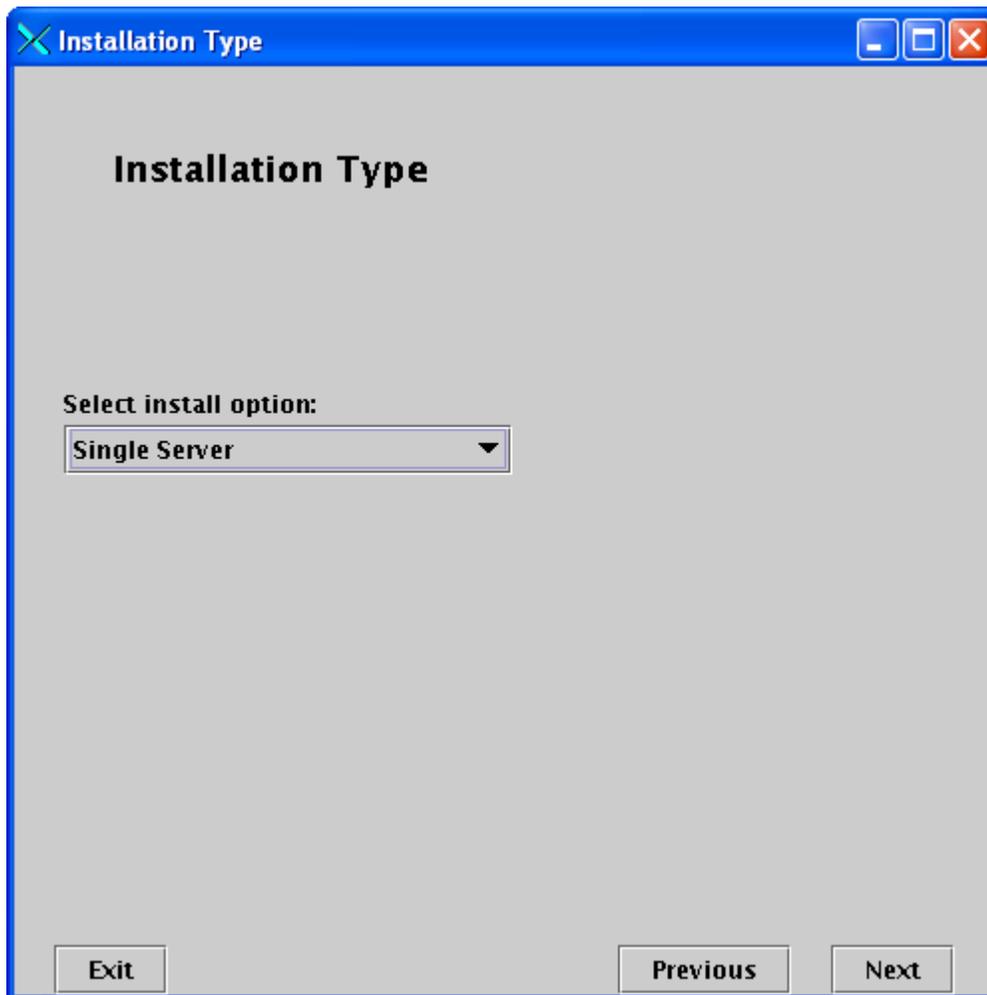
4. Enter the location of the `FWLicense.xml` file, which you received from FatWire, and click **Next**.



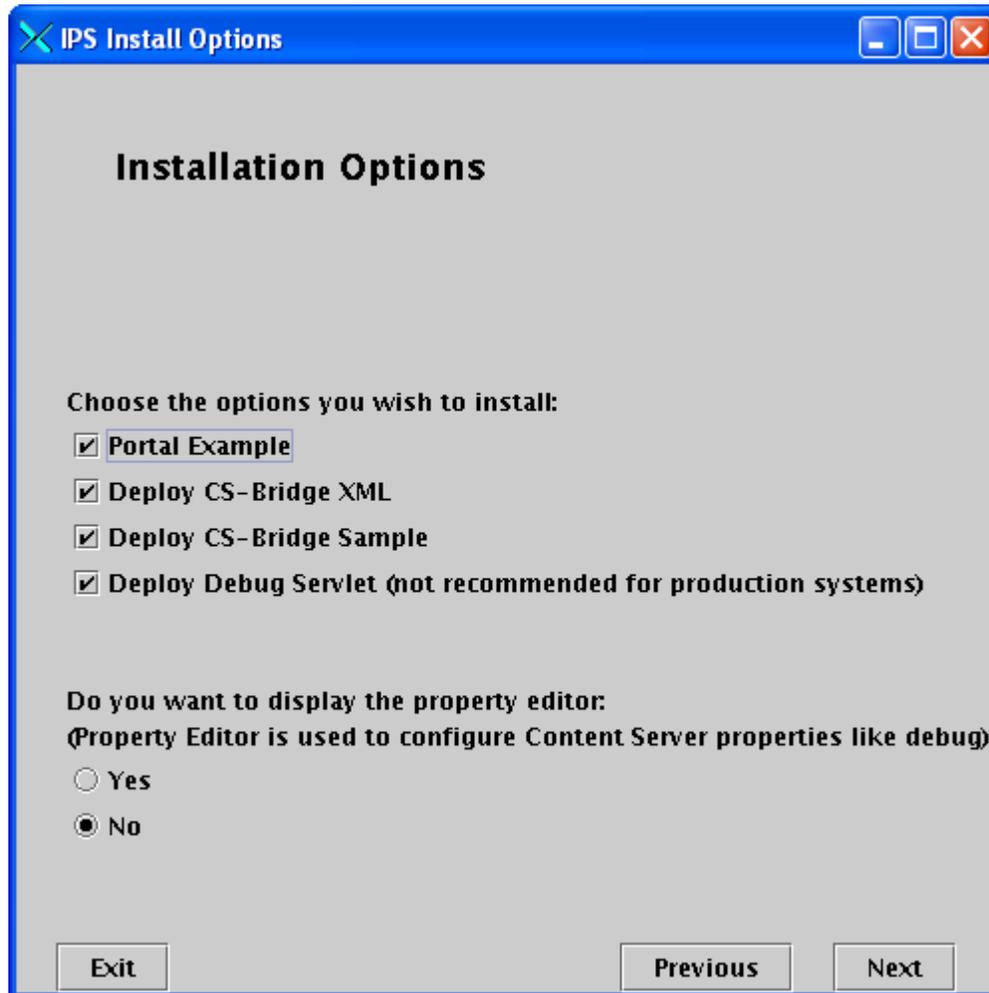
5. Select the products to install and click **Next**.



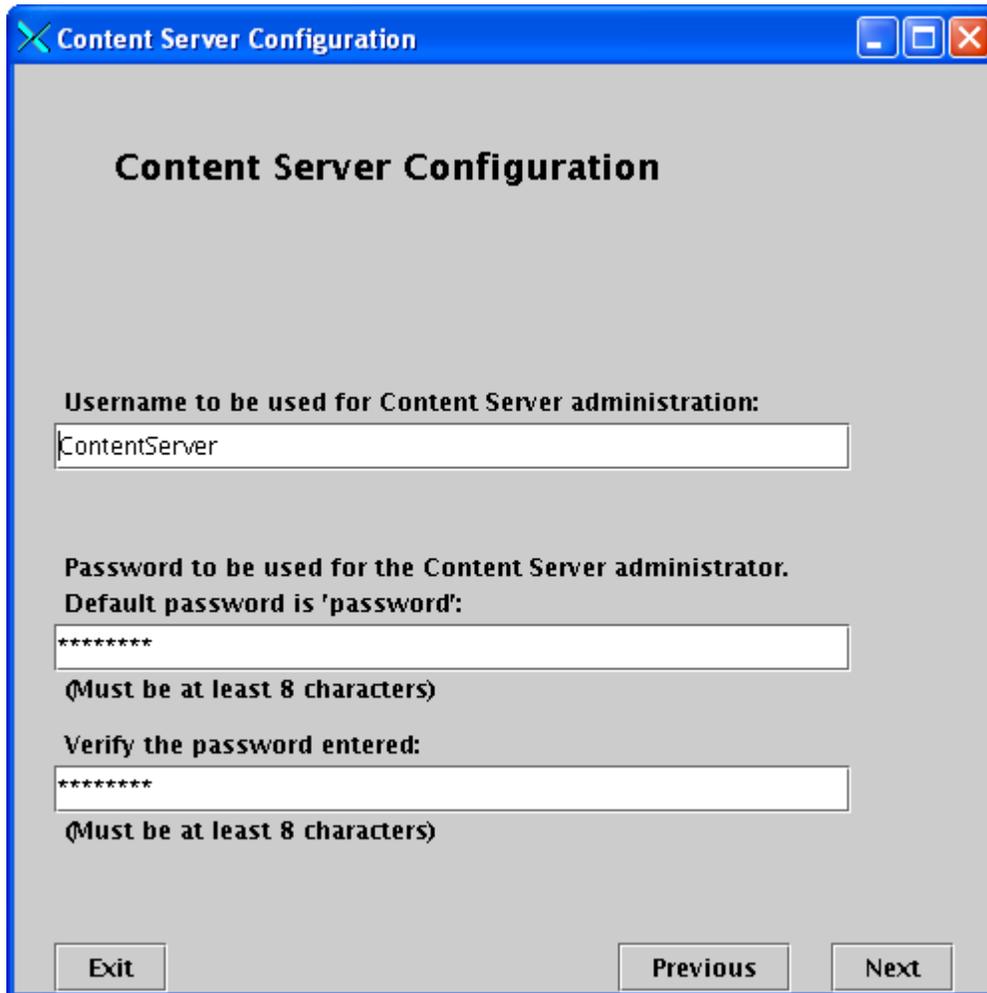
6. Select the **Single Server** installation type and click **Next**.



7. Select any additional installation options that apply (it is generally safe to use the defaults) and click **Next**.



8. Create a password for the Content Server user. (The default is `password`. It should be changed for security reasons.) When done, click **Next**.



The image shows a Windows-style dialog box titled "Content Server Configuration". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main area is gray and contains the following text and input fields:

**Content Server Configuration**

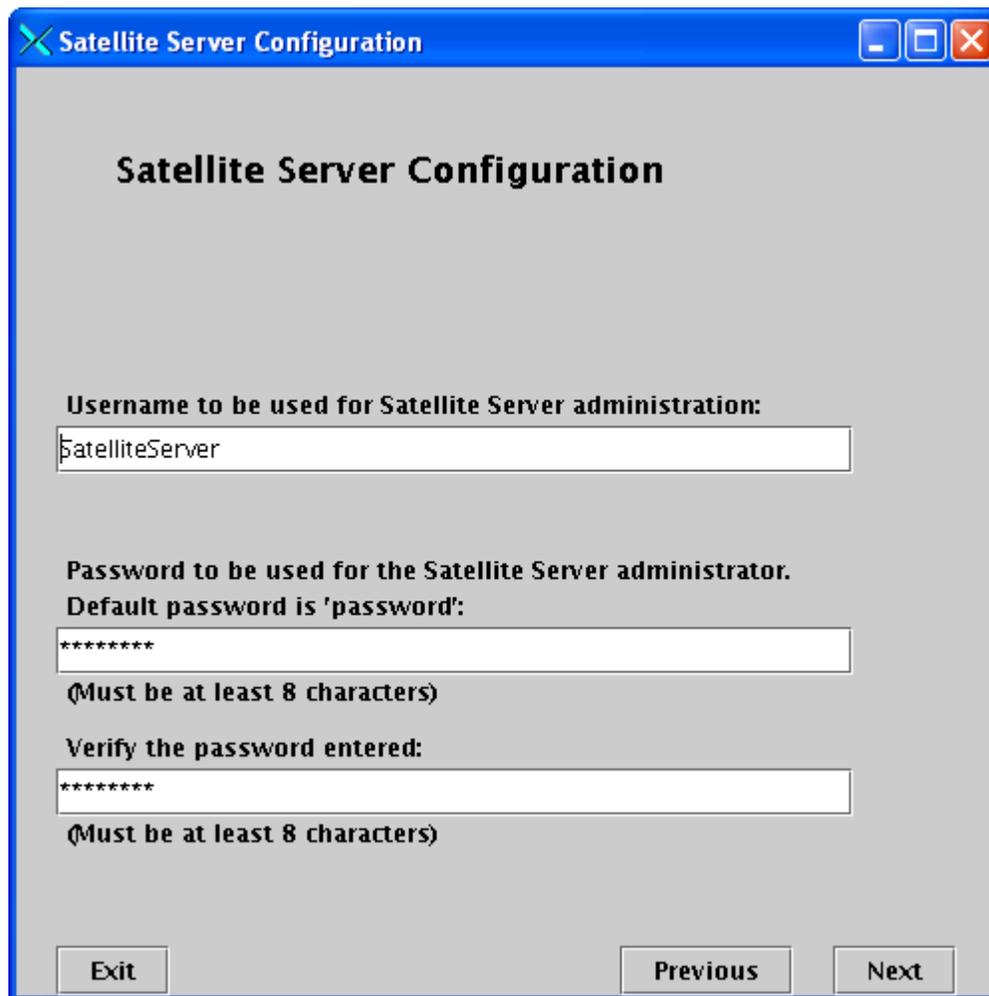
**Username to be used for Content Server administration:**

**Password to be used for the Content Server administrator.  
Default password is 'password':**  
  
(Must be at least 8 characters)

**Verify the password entered:**  
  
(Must be at least 8 characters)

At the bottom of the dialog, there are three buttons: "Exit" on the left, "Previous" in the center, and "Next" on the right.

9. Create a password for the Satellite Server user. (The default is `password`. It should be changed for security reasons.) When done, click **Next**.

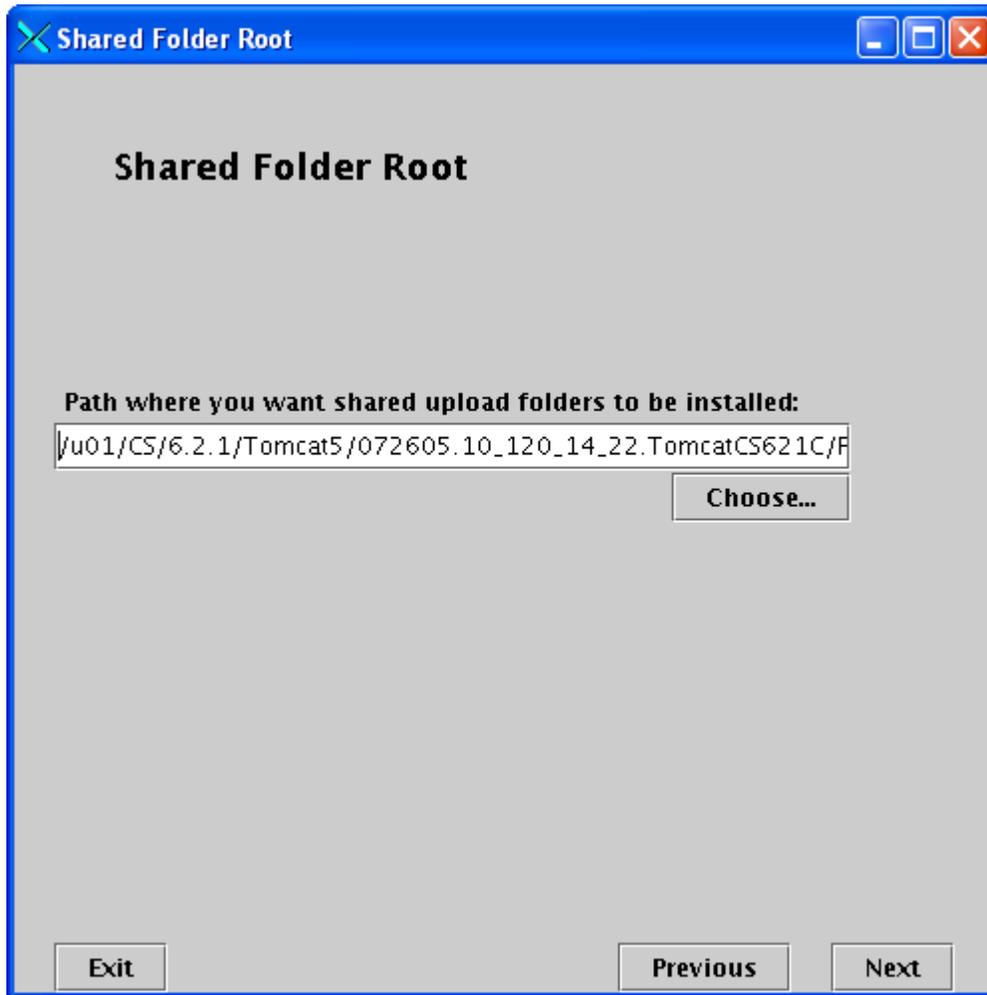


The screenshot shows a Windows-style dialog box titled "Satellite Server Configuration". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main content area is light gray and contains the following text and input fields:

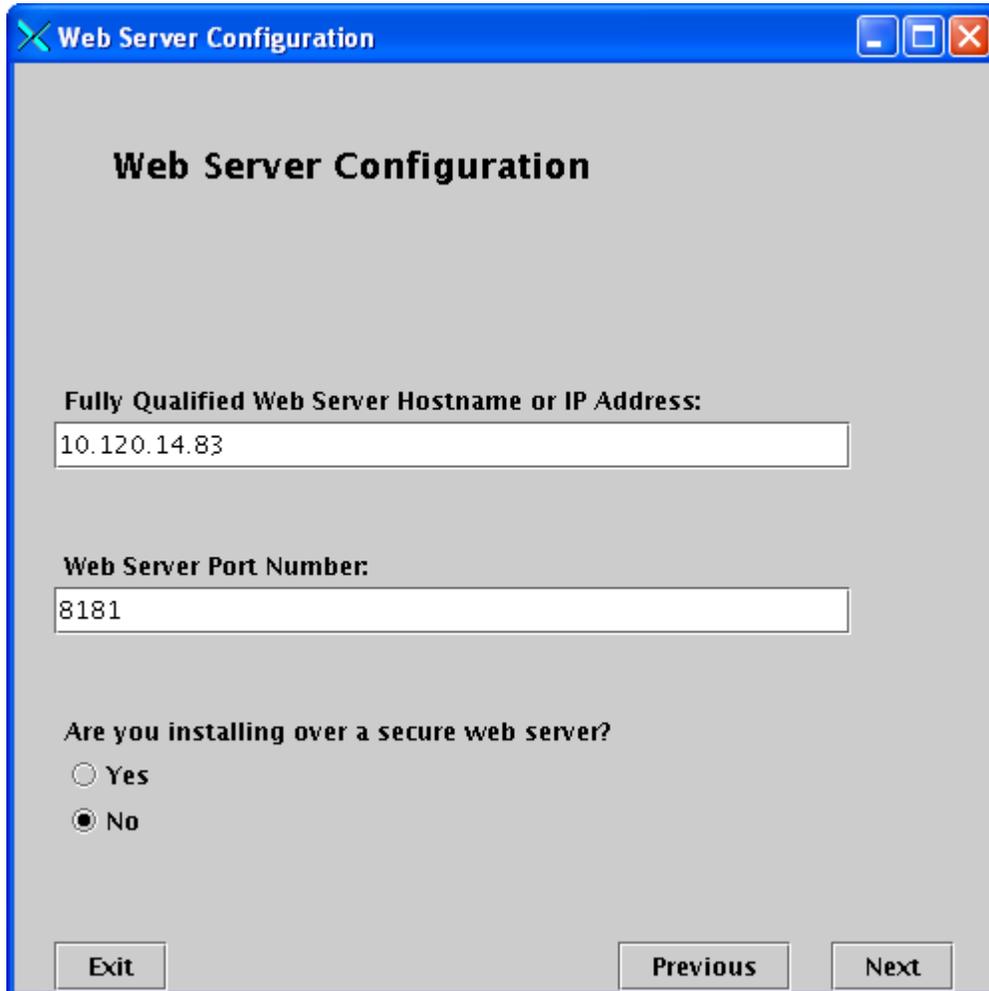
- Satellite Server Configuration** (Section Header)
- Username to be used for Satellite Server administration:**  
Input field containing "SatelliteServer"
- Password to be used for the Satellite Server administrator.**  
**Default password is 'password':**  
Input field containing "\*\*\*\*\*"  
(Must be at least 8 characters)
- Verify the password entered:**  
Input field containing "\*\*\*\*\*"  
(Must be at least 8 characters)

At the bottom of the dialog, there are three buttons: "Exit" on the left, "Previous" in the center, and "Next" on the right.

10. Enter the location of the shared file system that was created in the pre-installation phase and click **Next**.



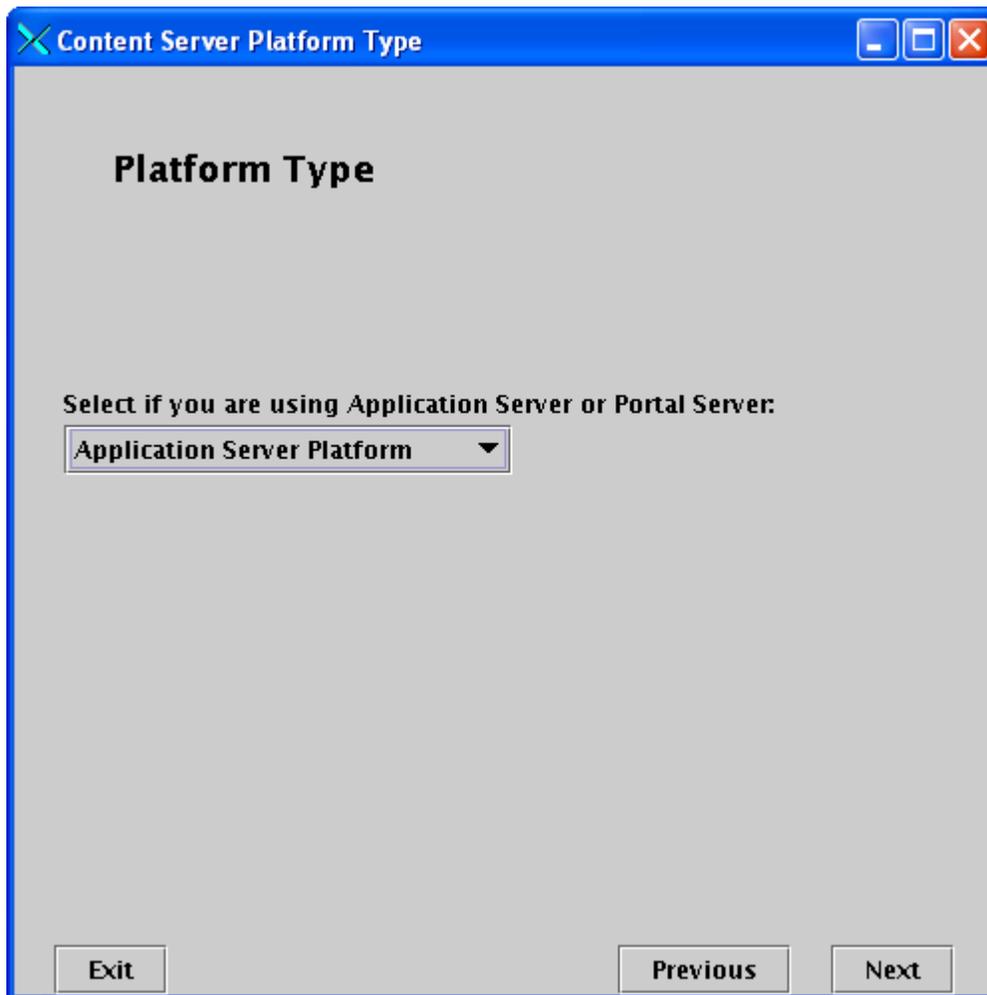
11. Enter the name and port of the host on which Content Server will reside. The port on which Tomcat is running is the main port replacing `<port>` in the `$CATALINA_BASE/conf/server.xml` file.



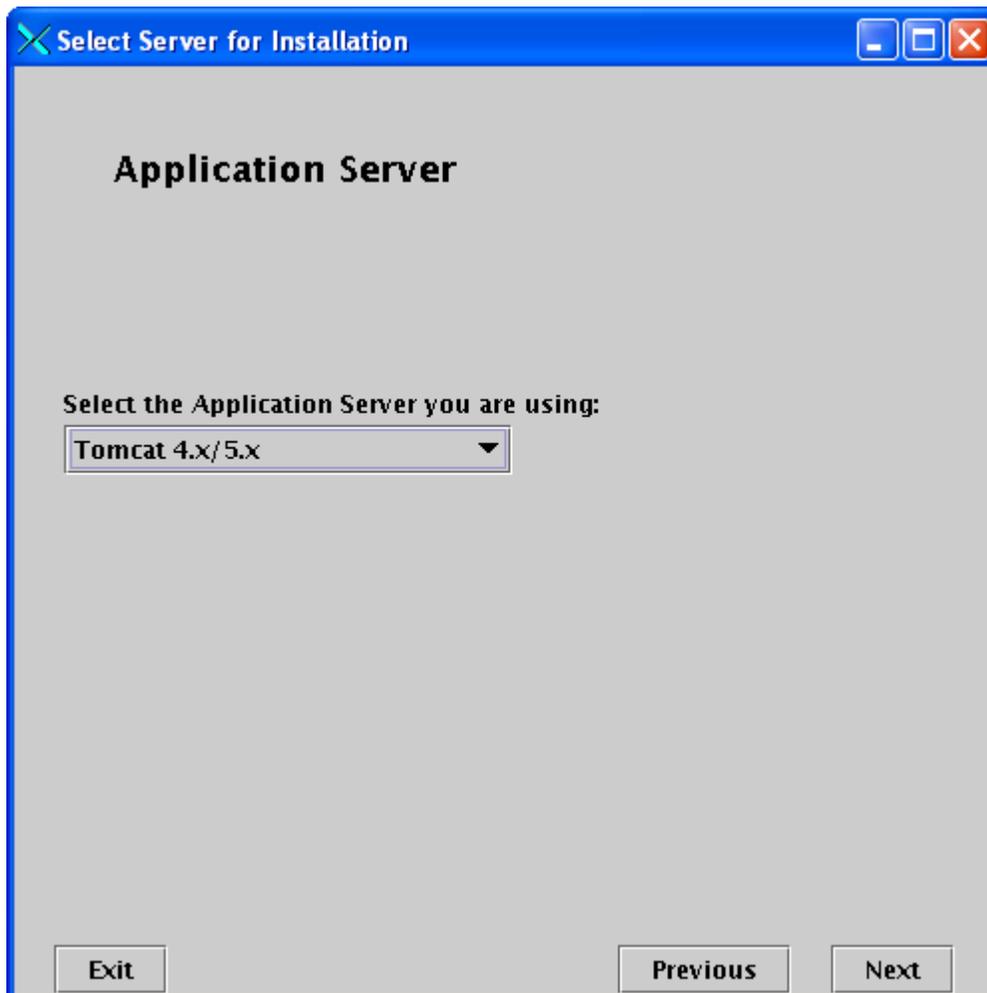
The image shows a Windows-style dialog box titled "Web Server Configuration". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main content area is light gray and contains the following elements:

- Web Server Configuration** (Section Header)
- Fully Qualified Web Server Hostname or IP Address:** A text input field containing "10.120.14.83".
- Web Server Port Number:** A text input field containing "8181".
- Are you installing over a secure web server?** A question with two radio button options: "Yes" (unselected) and "No" (selected).
- At the bottom, there are three buttons: "Exit" on the left, "Previous" in the center, and "Next" on the right.

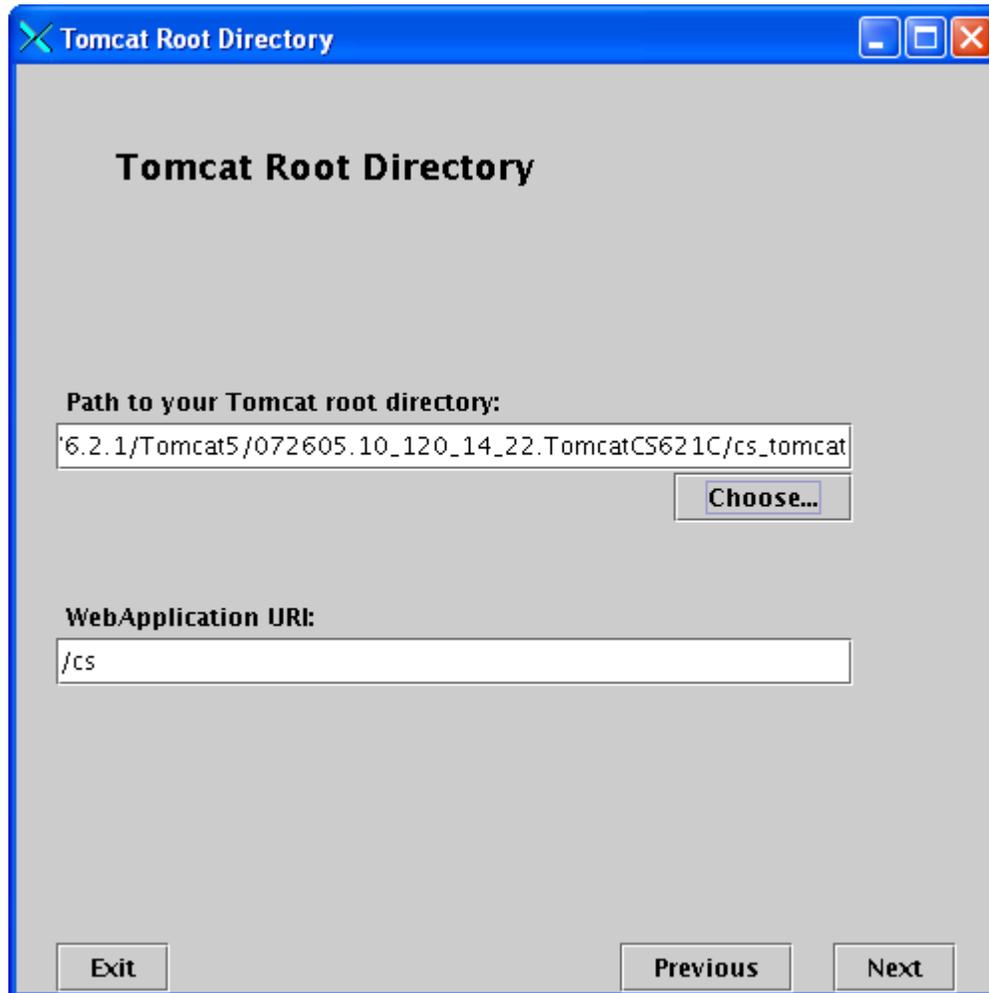
12. Leave the default of **Application Server Platform** selected and click **Next**.



13. Select **Tomcat 4.x/5.x** from the pull-down menu and click **Next**.



14. Specify the application deployment root directory (the value that is stored in `$CATALINA_BASE`). Leave the web application URL as `/cs`.



The screenshot shows a window titled "Tomcat Root Directory" with a blue title bar. The window has a grey background and contains the following elements:

- Tomcat Root Directory** (Section Header)
- Path to your Tomcat root directory:** A text input field containing the path `6.2.1/Tomcat5/072605.10_120_14_22.TomcatCS621C/cs_tomcat`. To the right of the field is a **Choose...** button.
- WebApplication URI:** A text input field containing the URI `/cs`.
- At the bottom of the window, there are three buttons: **Exit**, **Previous**, and **Next**.

**15. Configure the database:**

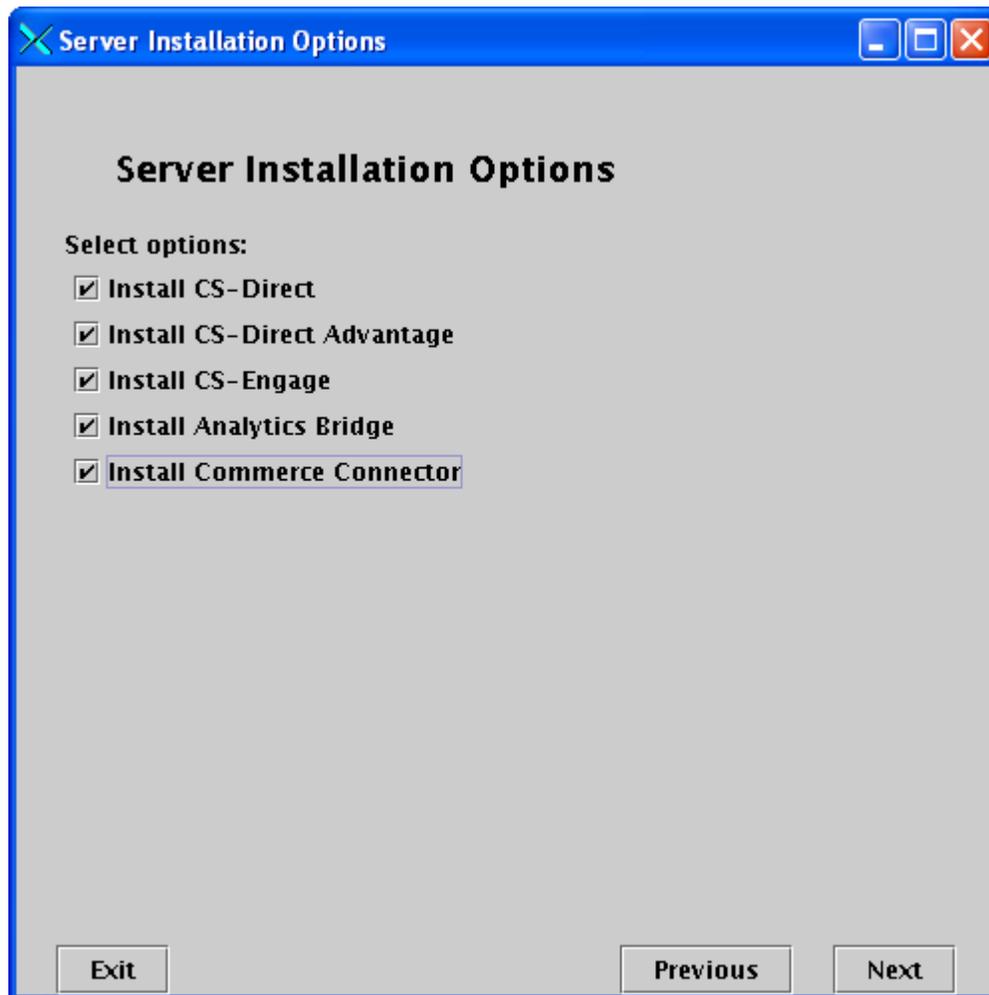
- a. From the drop-down menu, select the name of the database that will be used.
- b. Enter the name of your data source (this can be found in the `$CATALINA_BASE/conf/server.xml` file and, unless changed, is `csDataSource`).
- c. Click **Next**.



The screenshot shows a window titled "Database Configuration" with a blue title bar. The window contains the following elements:

- Database Configuration** (Section Header)
- Select the Database you are using:** (Label)
- (Drop-down menu)
- Enter JNDI Data Source Name: (This can be any value you choose)** (Label)
- (Text input field)
- Exit** (Button)
- Previous** (Button)
- Next** (Button)

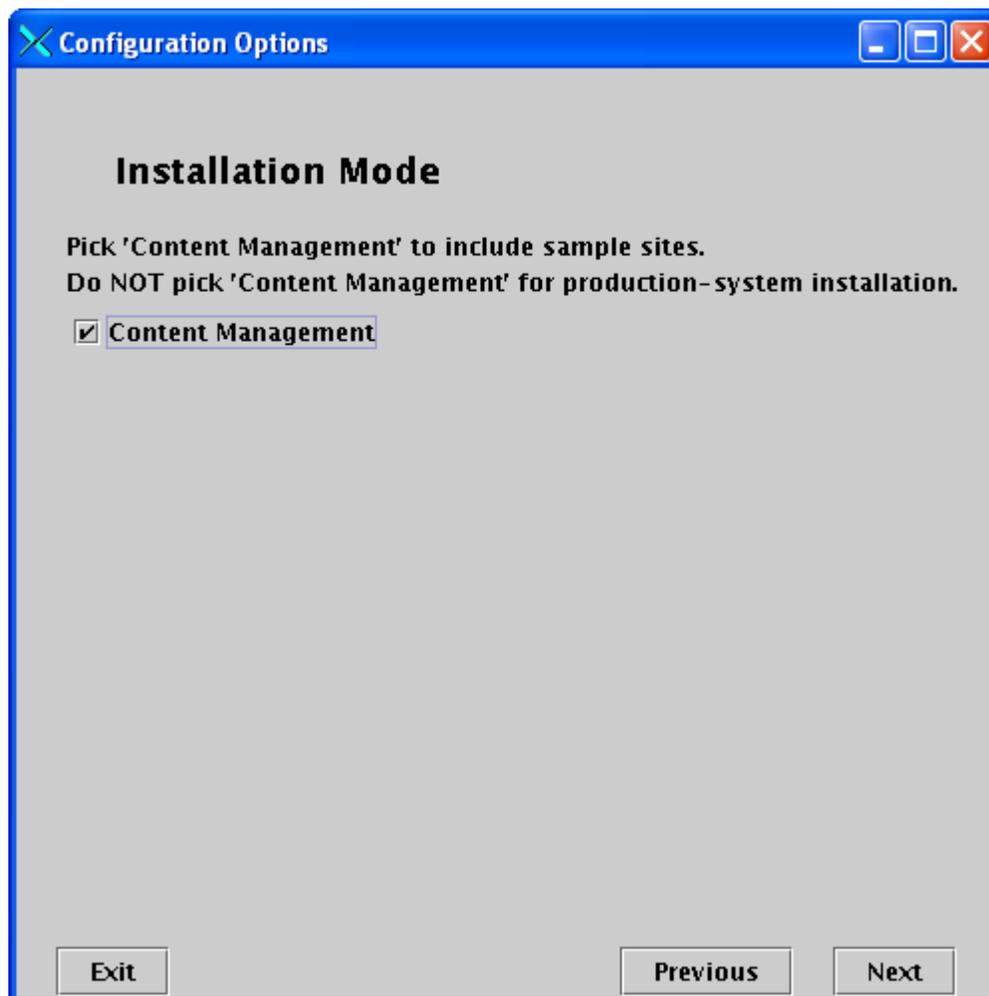
16. Select the components you have purchased and click **Next**.



17. Select the installation mode in order to bypass or accept the option of installing sample content and sample sites on the system you are setting up.

Do one of the following:

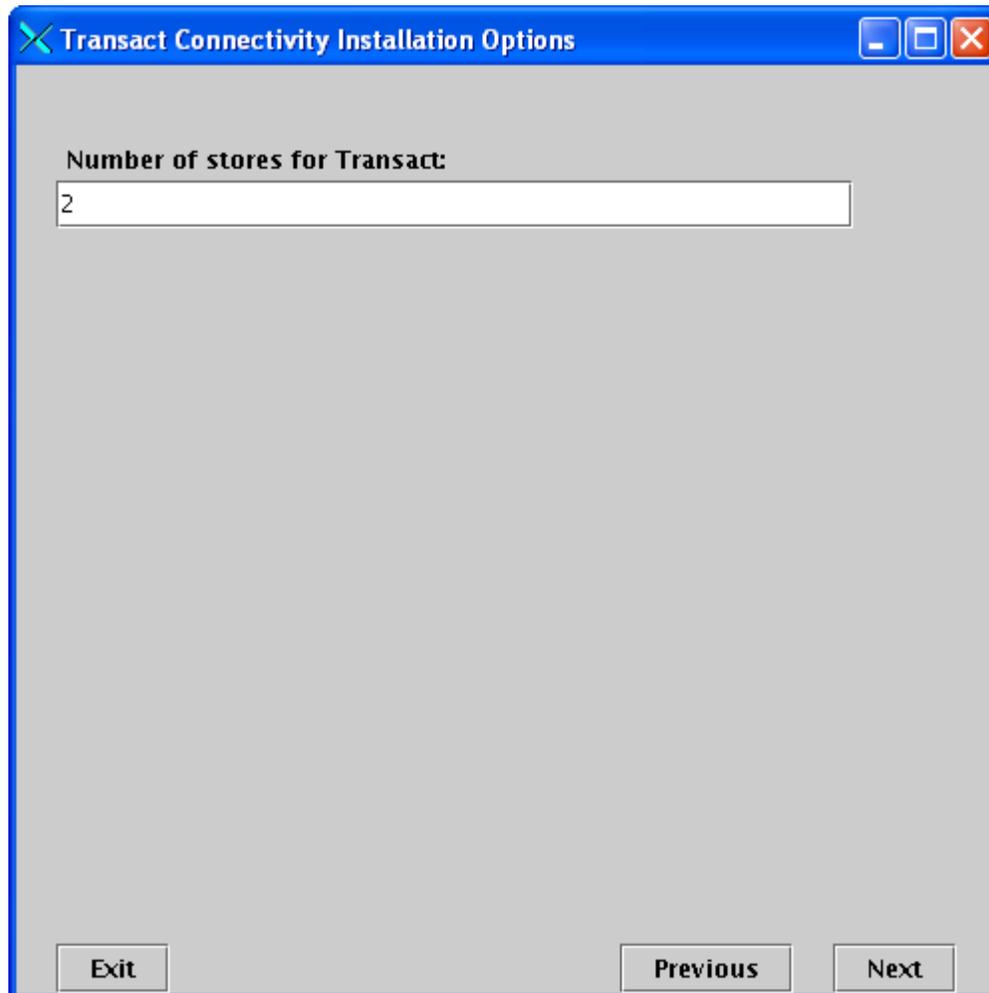
- Select **Content Management** if you are setting up a development or content management system, **and** you want to install sample content and sample sites on the system. Click **Next**, and select the samples you wish to have installed. When you see the “Transact Connectivity Installation Options” screen, continue with [step 18](#).
- Deselect **Content Management** if one of the following holds:
  - You do not want to install sample content and sample sites.
  - You are setting up a production system. (Sample content that might have been installed on the content management system will be mirrored to the publishing system during dynamic publishing.) Continue with [step 18](#).



**18. Set transact connectivity installation options:****Note**

The following screen is displayed only if you previously elected to install Commerce Connector.

Select the number of stores you wish Transact to have, then click **Next**.

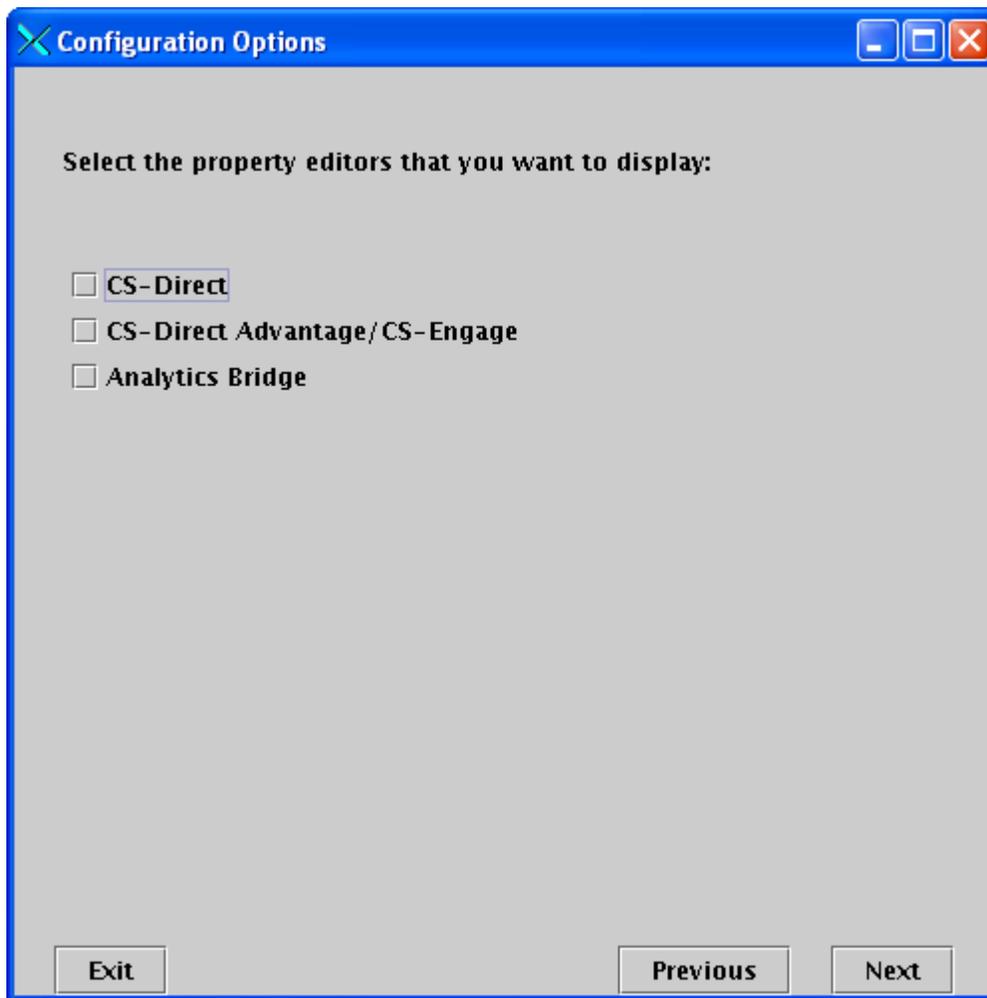


The screenshot shows a Windows-style dialog box titled "Transact Connectivity Installation Options". The dialog has a blue title bar with standard minimize, maximize, and close buttons. The main area is light gray and contains a label "Number of stores for Transact:" above a text input field. The input field contains the number "2". At the bottom of the dialog, there are three buttons: "Exit" on the left, "Previous" in the center, and "Next" on the right.

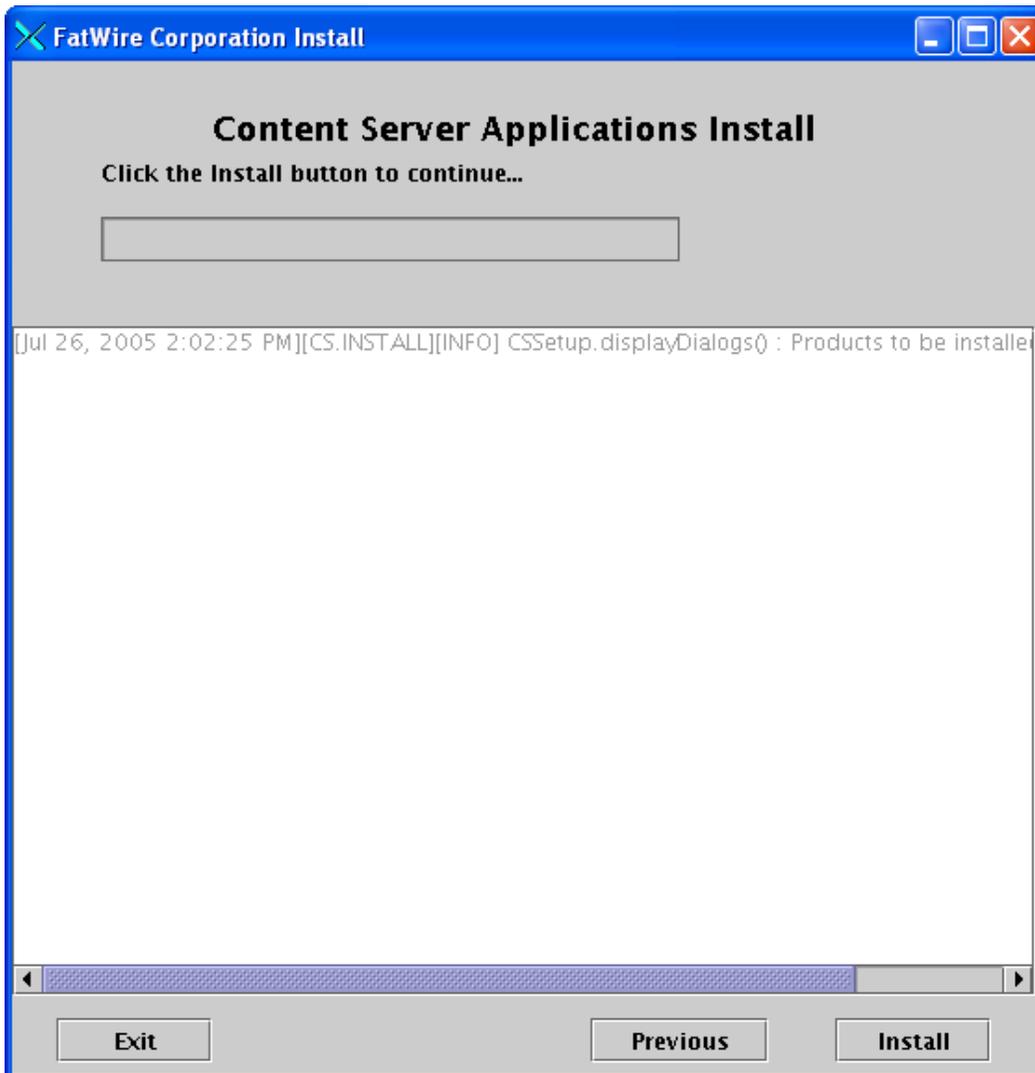
19. Select the content applications whose properties you want to edit during the installation, then click **Next**.

### Note

You can always view and modify the properties by using the Property Editor after the installation has been completed.

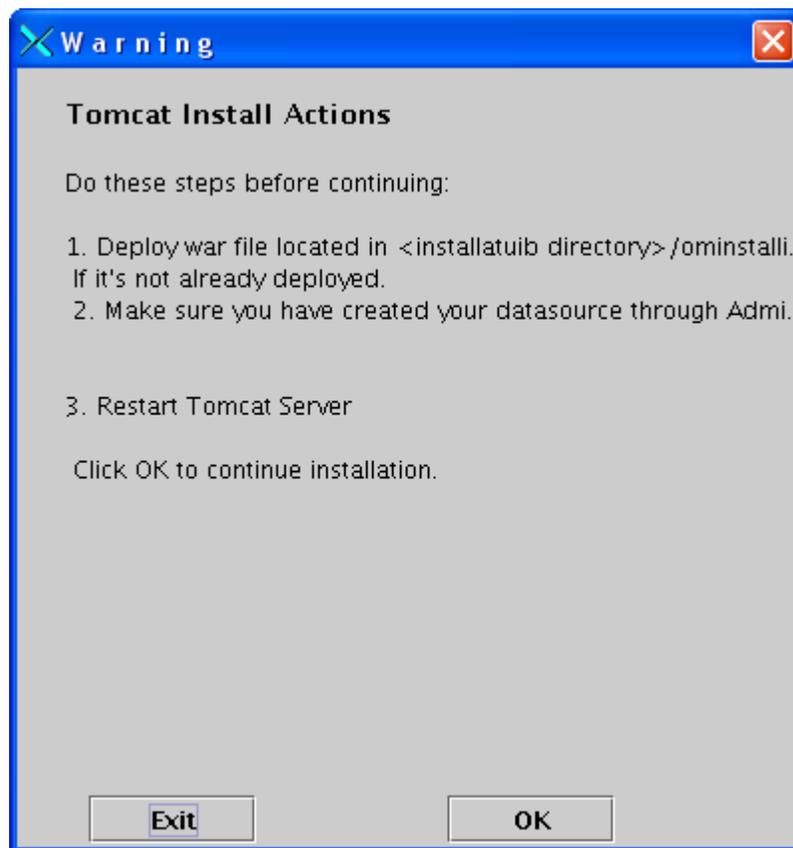


20. You are now ready to install Content Server.
  - a. Click **Install**.



**Note**

When the following pop-up window appears (halfway through the installation), it means that the Content Server base has been installed. You now need to deploy the application and test that it can connect to the database. Continue with [step b](#) to perform the operations requested by the popup window.



- b. Since Tomcat deploys the application automatically, the application only requires to be restarted with the following command:

```
$CATALINA_HOME/bin/shutdown.sh;sleep 5;$CATALINA_HOME/  
bin/startup.sh
```

- c. Once the application server has been restarted, test that the application is working and can connect to the database:

- 1) Test whether Content Server deployed successfully (use the values from the installation steps above):

```
http://<hostname>:<port>/cs/HelloCS
```

If this test fails, Content Server is unable to locate its components on the file system. Check your directory structure and restart the installation to copy files to their proper locations.

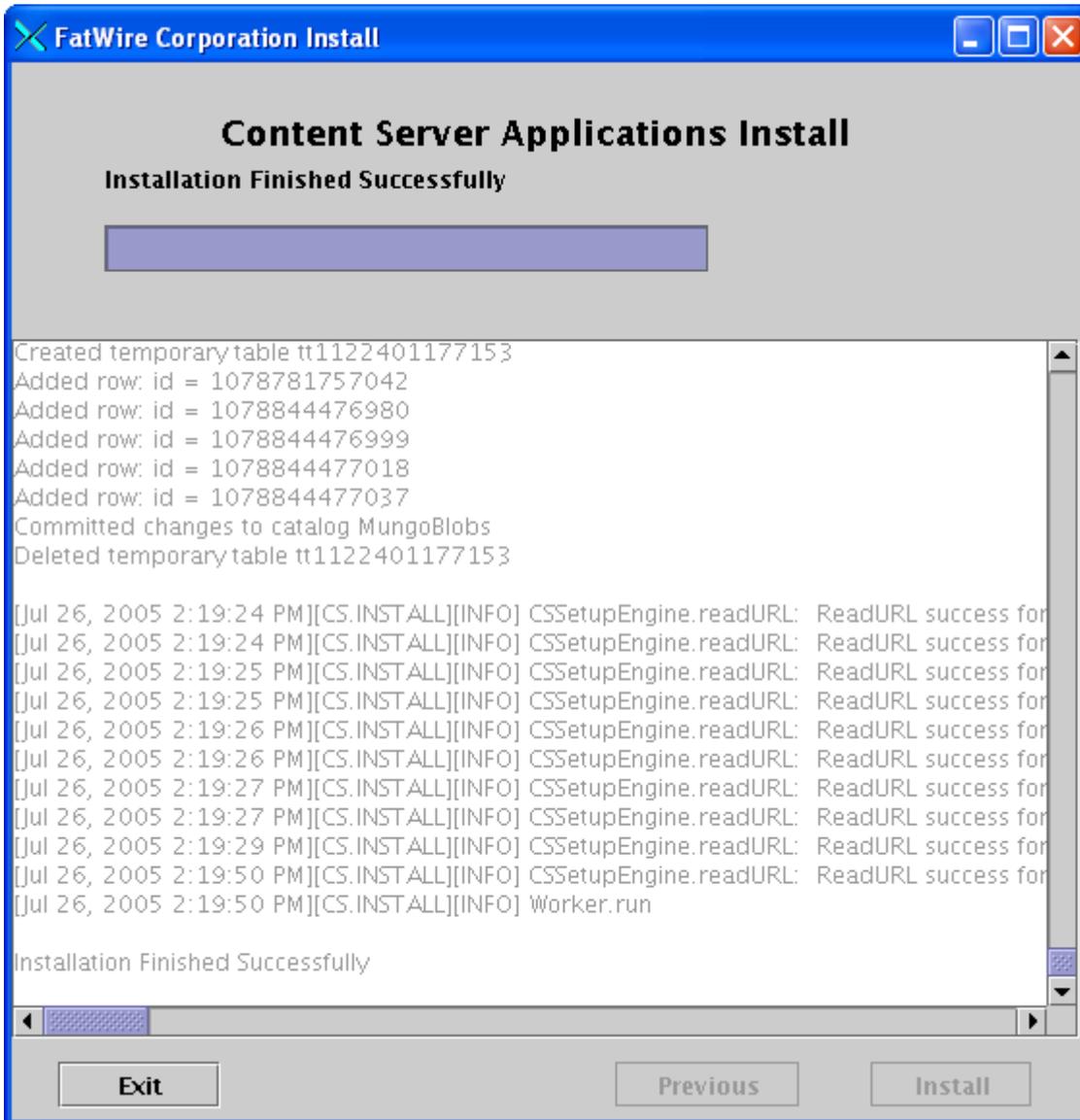
- 2) Test whether Content Server can communicate with the database:

`http://<hostname>:<port>/cs/Cataloganager?ftcmd=pingdb`

If this test fails, Content Server cannot connect to your database. Check your database to make sure it is running.

- d. If both tests are successful, you are ready to continue the installation. Click **OK**.

21. When the following screen is displayed, the installation is complete. Click **Exit** to quit the installer.



## Step III. Complete Post-Installation Procedures

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

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

Login Name: **fwadmin**

Password: **xceladmin**



2. If you plan to use the Verity search engine, follow the installation guidelines in [Appendix A, “Installing Verity Search Engine.”](#)

## Installing Content Server in a Clustered Environment

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

### To install Content Server in a clustered environment

1. Create a new Tomcat instance and Content Server installation directory by following the steps in [“Setting Up Directories,”](#) on page 17.
2. Change the port settings of the new instance by following the steps in [“Running Multiple Instances of Tomcat Simultaneously,”](#) on page 25.
3. Repeat the installation steps in [“Install Content Server,”](#) on page 38. Perform the steps exactly as you had in that procedure, with the exception of the following changes:
  - a. In the “Installation Directory” screen (see [step 3 on page 40](#)), select the installation directory you created in [step 1](#) of this section.
  - b. In the “Installation Type” screen (see [step 6 on page 43](#)), select **Cluster Member**.

- c. In the “Shared Directory Root” screen (see [step 10 on page 47](#)), select the shared directory root, which defaults to `<path to previous Content Server directory root>/Shared`.
  - d. In the “Web Server Configuration” screen (see [step 11 on page 48](#)), use the IP address and port of the server created in [step 2](#) of this section.
  - e. In the “Tomcat Root Directory” screen (see [step 14 on page 51](#)), enter the path to the installation directory created in [step 1](#) of this section.
4. For every member of the cluster, edit the `<path to cs_tomcat directory>/webapps/cs/WEB-INF/web.xml` file by adding the following line between the `<web-app>` and `<servlet>` lines:

```
<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 that belong to the same cluster must have the same values for `mcastAddr` and `mcastPort` in the Membership tag.
  - b. All Tomcat instances must have a different value for `tcpListenPort` in the Receiver tag.
6. Copy the `libFTFilelock.so` file to a location in the library path (usually `/usr/local/lib`):

```
cp <path to CS install directory>/bin/libFTFilelock.so
   /usr/local/lib
```

If you are running Linux, do the following (otherwise skip to [step 7](#)):

- a. Edit the `/etc/ld.so.conf` file by adding the following line:

```
<path to CS install directory>/bin
```
  - b. Run the `ldconfig` command.
7. Make sure there is a `usedisksync` directory in the Shared directory of your primary CS installation, or the first one that was created. If this directory is not already there, then create it with the following command:

```
mkdir <path to primary CS install directory>/Shared/
      usedisksync
```
  8. To finish configuring Content Server for clustering, you must edit the following properties using the CS Property Editor. Note that this must be done for all cluster members:
    - a. Launch the CS Property Editor (your display variable must be set):

```
<path to CS install directory>/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` should have a value of `false`
      - `ft.sync` should have a value assigned by you to all members of this cluster, e.g., `cluster1`.

- `ft.usedisksync` should be set to the path of the `usedisksync` directory created in [step 7](#).

## Load Balancing with `mod_jk`

### Note

To complete this section, you must have installed Tomcat Application Server with a Content Server cluster. If you have not already done so, complete the steps in [Chapter 4](#), “[Integrating Tomcat with Apache Web Server](#).”

1. For every member of the cluster, open the `server.xml` file and add the following attribute to the Engine tag:

```
jvmRoute="node1"
```

The instances can be given any node names as long as the names are unique.

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
```

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
```

3. Edit the `$APACHE2_HOME/conf/httpd.conf` file by changing the following line:  
    `JkMount /cs/* tomcat`  
to read  
    `JkMount /cs/* loadbalancer`
4. Load balancer configuration is now complete. Restart Apache for the changes to take effect.



## Appendix A

# Installing Verity Search Engine

The Verity Search Engine comes with a set of installation notes; however, a few changes and clarifications need to be made for it to work properly with Tomcat. The instructions below for installing the Verity Search Engine supplement the instructions in the Verity product documentation.

1. Copy the `libFTVeritySearch.so` file into the `$CATALINA_HOME/common/lib` directory.
2. Copy the jar file `Verityse.jar` into the third-party jar location:  
`$CATALINA_HOME/common/lib`
3. Edit the file `$CATALINA_HOME/startup.sh` as follows:

After the section dealing with setting `LD_LIBRARY_PATH`, add the following two lines:

```
# verity Additions
CSVERITYPATH=<content server installation directory>/VerityK2
 / <_platform>
LD_LIBRARY_PATH=$LD_LIBRARY_PATH:$CSVERITYPATH/
filters:$CSVERITYPATH/bin
```

