Project WebSynergy Milestone 4 Administration Guide

Beta

Sun Microsystems, Inc.
4150 Network Circle
Santa Clara, CA 95054
U.S.A.

Part No: 820–6878

090121@21808

Contents

Preface

Project WebSynergy is Sun's next-generation Web 2.0 application aggregation and presentation platform. It is a suite of integrated software products that enables businesses large and small to pull together applications and content from a variety of Web-based and internal sources and present them as a unified, customizable portal on Web browsers, kiosks, and mobile devices. WebSynergy makes it easier for users to find and use the tools and information they need when they need them.

Project WebSynergy is a collaborative effort between Sun Microsystem's Portal Server team, the Liferay Community, and the OpenPortal community. WebSynergy integrates technologies from these three products to enable enterprise-class portals that are easy to use for end users, system administrators, and developers alike.

WebSynergy enables:

- **Social Networking for the Enterprise**

  Find people with the answers across organizations, silos, and hierarchies, and boost productivity in distributed teams.
- **Enterprise Widgets for Collaboration Work**

  Bundled Productivity Widget Suites to get work done (CMS, Workflow), and Information Widget Suite for Effective Collaboration (blogs, wikis).
- **Internet Widgets on the Enterprise Extranet**

  Build stickiness into your intranet with Google Gadgets, YouTube, and Facebook widgets.

Who Uses Project WebSynergy?

Project WebSynergy offers features for general users, system administrators, and application and portal developers.

- For **General Users**, WebSynergy makes it easy to communicate, collaborate, and customize the applications shared by your organization. Social networking components, like blogs, wikis, bookmarks, and messaging can be presented side-by-side with proprietary and third-party business and productivity applications. Different user communities can have their own portals, layouts, and customizations, and WebSynergy can be scaled down or up to suit organizations from the very small up to the enterprise level.

- For **System Administrators**, WebSynergy makes it easy to manage users, groups, communities, permissions, and highly specific levels of security. A browser-based GUI makes managing portals, portlets, plugins, and applications as simple as dragging and dropping. WebSynergy runs on top of the enterprise-class, open source GlassFish Application Server, which means that WebSynergy can be optimized according to your needs for performance, reliability, security, load balancing, and clustering, among other server characteristics.
- For **Application and Portal Developers**, WebSynergy makes it easy to develop portlets, portals, plugins, services, content management, workflows, and themes using your own tools, including NetBeans™, Eclipse, and Dreamweaver. WebSynergy is based on the open source OpenPortal and Liferay 5.2 code bases, which means what you develop in WebSynergy will be standards-based, portable, and maintainable. You can combine familiar Ajax, Jmaki, Ruby, PHP, and Java technologies and techniques with WebSynergy's powerful presentation capabilities to deliver the kind of rich, dynamic, interactive user experience previously available only in complex custom-built Web applications.

Project WebSynergy Features

Project WebSynergy provides a new class of portal functionality by which users can define their own Web spaces. With built-in content and document management, human workflow development tools, enterprise identity integration, and social networking features, system administrators and application developers can now deploy a platform that allows for rapid rollout of next generation Web capabilities for their users.

Key WebSynergy features include:

- Identity-based content delivery
- User-managed Web spaces, including user self-Web publishing and user access controls
- Rapid and simple Web site design tools make it possible to quickly set up Web sites for content, collaboration, and interactivity, with out of the box templates
- Built-in collaboration, with bundled wiki, blog, task management, calendaring, document sharing, polls, IM, and email applications
- Document and Web content management system with workflow, staging, and publication tools
- Syndicate portlets and widgets with REST
- Develop widgets using the JSR 286 portlet standard as well as scripting standards such as Ruby, PHP, and JMaki

Before You Read This Book

Readers should be familiar with the following products and concepts:

- Sun Java System Directory Server
- Sun Java System Access Manager
- Your web container
  - Sun GlassFish Enterprise Server v2/v3
- Your operating system
- Basic UNIX administrative procedures
- LDAP (lightweight directory access protocol)
- Web Services for Remote Portlets (WSRP)

How This Book Is Organized

The following are the chapters in the book:

1. Chapter 1, "Administration Overview"
2. Chapter 2, "Installation"
3. Chapter 3, "Customizing Project WebSynergy"
4. Chapter 4, "Administering Roles and Permissions"
5. Chapter 5, "Advanced WebSynergy Configuration"
6. Chapter 6, "Administering Content Management System"
7. Chapter 7, "Sample Use Cases for WebSynergy Portlets"
8. Chapter 8, "Web Services for Remote Portlets"
9. Chapter 9, "System Maintenance"
10. Chapter 10, "Performance Tuning"
11. Chapter 11, "Troubleshooting"

Related Project WebSynergy Documentation

The Project WebSynergy documentation set is available on the Project WebSynergy Document Collection page.

Additional documentation is also available on the Liferay wiki, Liferay Community Documentation, and OpenPortal Documentation sites.

Related Third-Party Web Site References

Third-party URLs are referenced in this document and provide additional, related information.

Note – Sun is not responsible for the availability of third-party web sites mentioned in this document. Sun does not endorse and is not responsible or liable for any content, advertising, products, or other materials that are available on or through such sites or resources. Sun will not be responsible or liable for any actual or alleged damage or loss caused or alleged to be caused by or in connection with use of or reliance on any such content, goods, or services that are available on or through such sites or resources.

Documentation, Support, and Training

The Sun web site provides information about the following additional resources:

- Documentation (`http://www.sun.com/documentation/`)
- Support (`http://www.sun.com/support/`)
- Training (`http://www.sun.com/training/`)

Typographic Conventions

The following table describes the typographic conventions that are used in this book.

TABLE P–1 Typographic Conventions

Typeface	Meaning	Example
`AaBbCc123`	The names of commands, files, and directories, and onscreen computer output	Edit your `.login` file. Use `ls -a` to list all files. `machine_name% you have mail.`

`AaBbCc123`	What you type, contrasted with onscreen computer output	`machine_name%` **`su`**
		`Password:`

aabbcc123	Placeholder: replace with a real name or value	The command to remove a file is `rm` *filename*.

TABLE P–1 Typographic Conventions *(Continued)*

Typeface	Meaning	Example
AaBbCc123	Book titles, new terms, and terms to be emphasized	Read Chapter 6 in the *User's Guide*. A *cache* is a copy that is stored locally. Do *not* save the file. **Note:** Some emphasized items appear bold online.

Shell Prompts in Command Examples

The following table shows the default UNIX® system prompt and superuser prompt for the C shell, Bourne shell, and Korn shell.

TABLE P–2 Shell Prompts

Shell	Prompt
C shell	`machine_name%`
C shell for superuser	`machine_name#`
Bourne shell and Korn shell	`$`
Bourne shell and Korn shell for superuser	`#`

CHAPTER 1

Administration Overview

This chapter is helpful to gain a basic understanding of Project WebSynergy.

WebSynergy Terminology

The following are some common terms that are useful to know.

Portlet Portlets are pluggable user interface software components that are managed and displayed in a web portal. Typically, a portal page is displays as a collection of non-overlapping portlet windows, where each portlet window displays a portlet. WebSynergy portlets confirm to JSR286 standards.

Instanceable Portlet If you can add many instances of a portlet to your page, the portlet is called an instanceable portlet.

Non-instanceable Portlet If you can add just one instance of a portlet to your page, the portlet is called a it is a non-instanceable portlet.

Public Page and Private Page Every individual user account in a WebSynergy site contains public pages and private pages. Pages that can be accessed by a guest user are public pages. Pages that can only be accessed by logging in to a user account are private pages.

Widget Widgets are similar to portlets, except that unlike portlets, widgets are created using a programming language other than Java, such as PHP or Ruby.

Friendly URL The modifiable part of the URL for the private page of an user account.

Theme The "look and feel" settings that are applied to a portal page. Themes can be applied sitewide by a WebSynergy site administrator or on a page-by-page basis by registered users.

Layout Template A scheme that defines the general layout of portlets and widgets on a portal page. As with themes, layout templates can be applied by site administrators and registered users.

Guest A user who does not have an account on a given WebSynergy site. Typically, guest users have limited ability to view or modify content on a WebSynergy site.

User A person who is registered on a WebSynergy site.

User Group A grouping of users; members of a user group typically share common sets of access permissions.

Community A collection of users sharing common interests or goals; community members typically share a common set of portlets and pages. For example, a community might have a wiki that is not available to users outside the community. Multiple user groups can coexist within a given community.

Organization A hierarchical collection of users, user groups, and communities. Multiple communities can coexist within a given organization.

User Role Permissions and access rights defined for a given user; typically maps to rights within groups, communities, and organizations.

Creating an User Account on WebSynergy

Any user who can accesses the HTTP instance of WebSynergy can create a user account on it. You can sign in by selecting *Sign In* from the welcome menu. If you are an existing user, you can sign in by using your user name and password. To create a new account click on the *Create Account* link and follow the instructions to create an account. When you add an user account, the user is added to the list of Users in the Enterprise Admin Portlet. The admin user can monitor all the user accounts.

Also, you can login to WebSynergy by using an OpenID. To be able to login using an OpenID, you need to get an OpenID from any of the popular OpenID providers. For more information, see `http://openid.net/`. You can get an OpenID from any of the popular OpenID providers such as, are claimID, myOpenID, VeriSign's Personal Identity Provider, myID.net, and myVidoop.

When you create an OpenID with a OpenID provider such as myOpenID, you get an email to the mail ID provided by you. Your OpenID will be authenticated when you check your mail, and you can use the OpenID to login to any of the websites which support OpenIDs. The OpenID is in the form of an URL, and you can sign-in and sign-out of the OpenID. When you have signed into an OpenID, you can login to any of the websites which support OpenIDs by using your OpenID.

The below procedure discusses the example of signing into WebSynergy by using an OpenID created by using myOpenID.

▼ To login to WebSynergy Using an OpenID

1. **Access the HTTP instance for WebSynergy using** `http://`*<machine-name>*`:8080`**.**

2. **Choose Sign In from the welcome menu.**

3. **Click on the OpenID link.**

4. **Specify your OpenID and click Sign In.**

   In this example, specify the OpenID from myOpenID. You need to be logged on to your OpenID for the OpenID authentication to work.

5. **Disable the Skip this step next time I sign in to** `http://`*<machine-name>*`:8080` **option, and click Continue.**

6. **Provide the required information and click Save.**

   Note – For the OpenIDs provided by yahoo and many other OpenID providers, you need to register the OpenID by selecting **My Account –> Identification –> OpenID**.

Defining Password Policies for Users

The admin user can use the *Enterprise Admin* portlet to define password policies for Users and Organizations. You can change the default password policy or can create a new password policy, and can apply different password policies to different sets of users and organizations.

▼ To Define a Password Policy

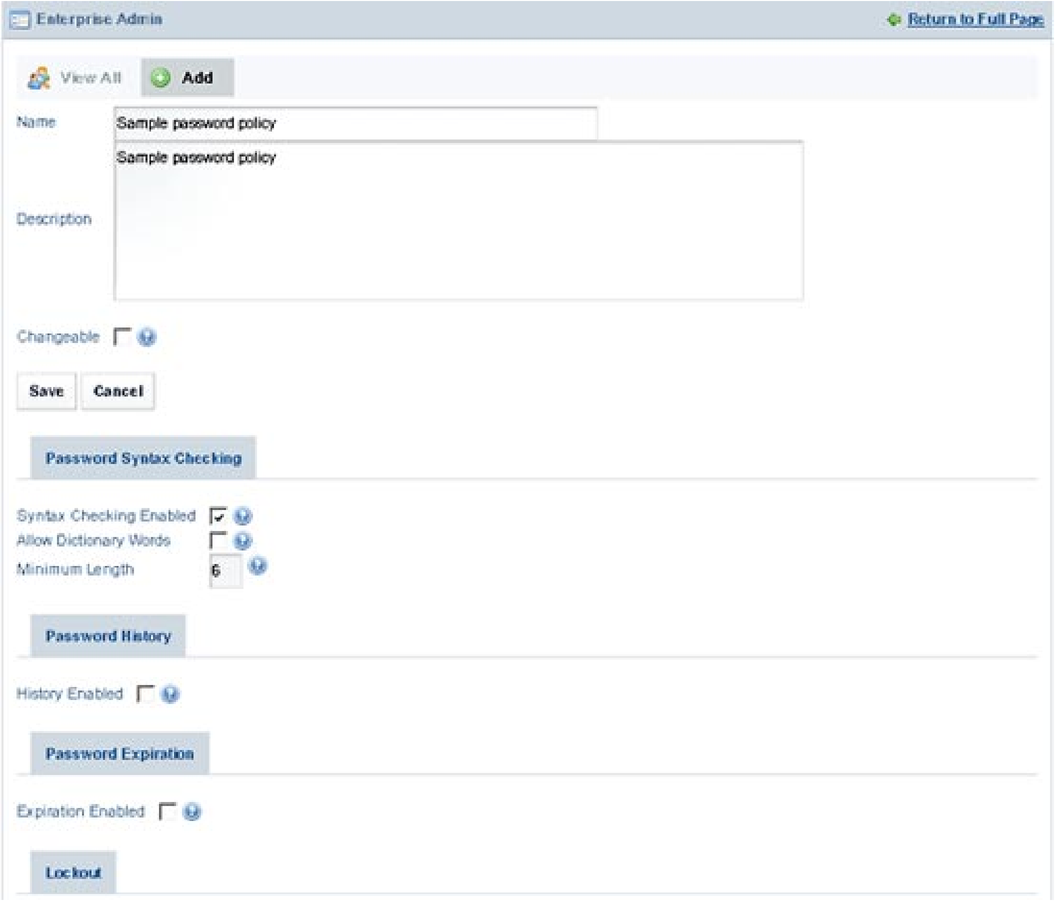
1. **Login to WebSynergy as Admin User.**

2. **Click on the Password Policies tab in the Enterprise Admin portlet.**

   You need to expand the portlet to access the Password Policies tab.

3. **Click on the Add button to add a new password policy.**

   Specify a name and description. You can enable some of the options as per your requirement. To know what an option is, place the cursor on the help button next to the option.



Name: Sample password policy

Description: Sample password policy

Changeable ☐

Password Syntax Checking

Syntax Checking Enabled ☑
Allow Dictionary Words ☐
Minimum Length 6

Password History

History Enabled ☐

Password Expiration

Expiration Enabled ☐

Lockout

FIGURE 1–1 Defining a password policy

4 **Click Save and return to the full screen.**

5 **To assign users or organizations to the password policy, click on the Actions button against the password policy and select Assign Members from the menu.**

Using the Welcome Menu for Administration Tasks

To be done.

▼ To use the Welcome Menu

1 **Login to WebSynergy as Admin User.**

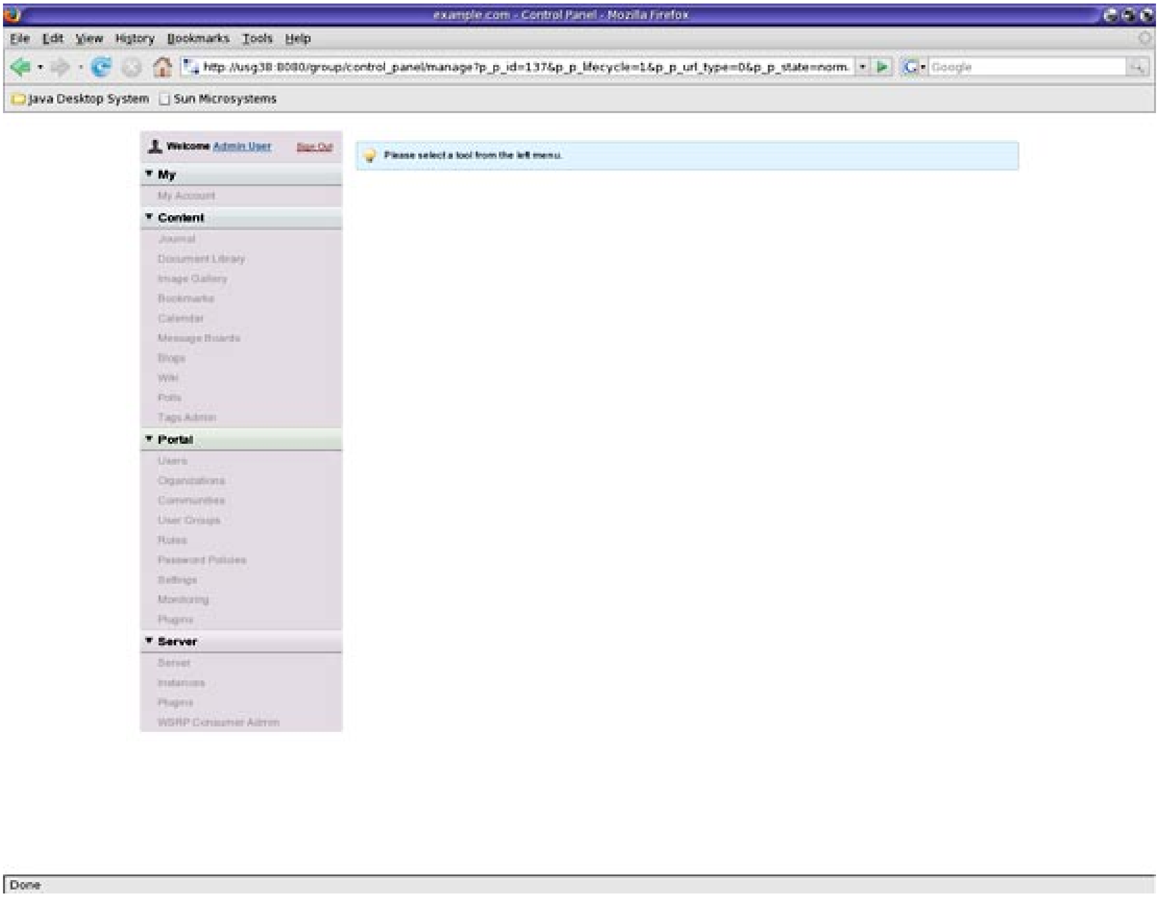
2 **Navigate to the Welcome menu, and click on the menu item that you need to access.**

Home, *Control Panel*, *My Account*, *Sign Out*, *Add Application*, *Layout Template*, *Manage Pages*, *Toggle Edit Controls*, and *My Places* are the items which can be accessed from the welcome menu for the admin user.

"Control Panel" on page 23	To apply control settings
"My Account" on page 25	To apply account settings
"Add Application" on page 26	To add portlets and widgets to your page
"Layout Template" on page 27	To change the layout template
"Manage Pages" on page 28	To specify a friendly URL to your public and private pages
"Toggle Edit Controls" on page 28	To toggle edit controls of the portlets on your page
"My Places" on page 29	To navigate to public and private pages associated with your account

Control Panel

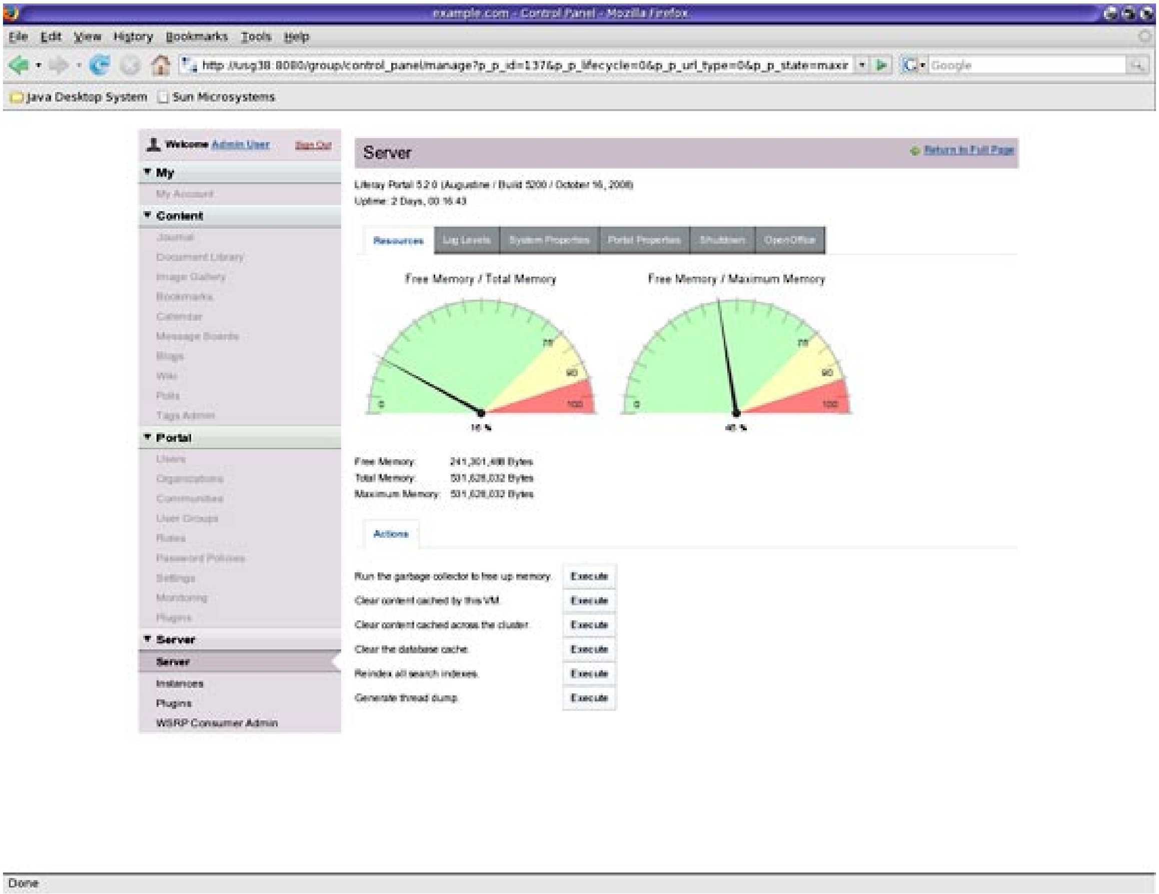
You can use the Control Panel to change the settings applied to *My Account*, *Content*, *Portal*, and *Server settings*. Select *Control Panel* from the main menu. The following page appears:



Please select a tool from the left menu.

FIGURE 1–2 Control Panel

You can select a tool from the left menu to apply settings to it. For example, click on Server under the Server menu.



Liferay Portal 5.2.0 (Augustine / Build 5200 / October 16, 2008)
Uptime: 2 Days, 00:16:43

Free Memory: 241,301,408 Bytes
Total Memory: 531,628,032 Bytes
Maximum Memory: 531,628,032 Bytes

Run the garbage collector to free up memory. Execute
Clear content cached by this VM. Execute
Clear content cached across the cluster. Execute
Clear the database cache. Execute
Reindex all search indexes. Execute
Generate thread dump. Execute

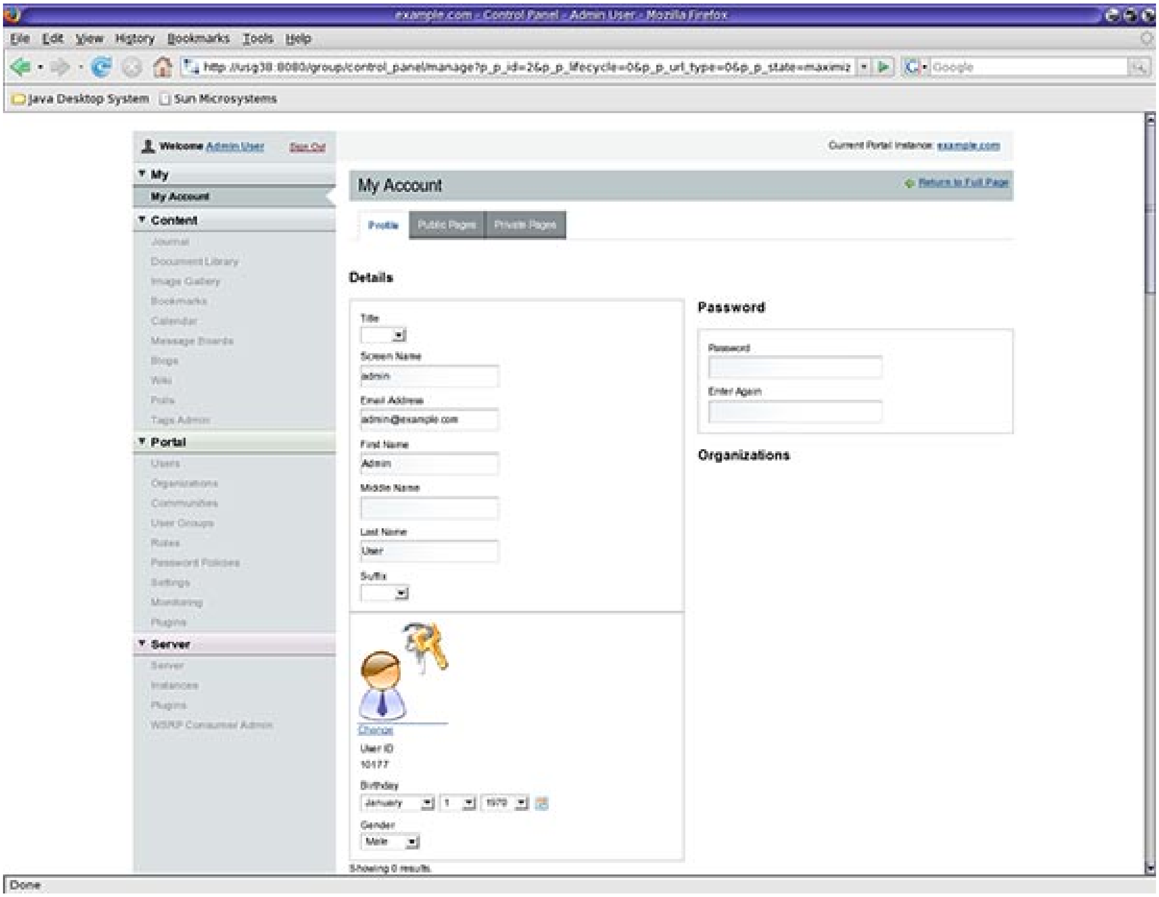
FIGURE 1–3 Control Panel

You can perform a number of server maintenance activities from this page.

Similarly, you can apply different kind of settings by using the Control Panel. You can click on the Welcome User and Sign Out links in the top left corner of the page, respectively to return to the main page and to sign-out of WebSynergy.

My Account

To access the *My Account* page, you can click on 'My Account' for the *Control Panel* window or you can choose it from the welcome menu.



http://usg38:8080/group/control_panel/manage?p_p_id=2&p_p_lifecycle=0&p_p_url_type=0&p_p_state=maximiz

Title:

Screen Name: admin

Email Address: admin@example.com

First Name: Admin

Middle Name:

Last Name: User

Suffix:

User ID: 10177

Birthday: January 1 1970

Gender: Male

Password:

Enter Again:

Showing 0 results.

FIGURE 1–4 My Account

From this page, you can view and make changes to Public Pages and Private Pages associated with your account. Also, you can view and make changes to your profile.

Add Application

You can add a portlet or a widget to your page by clicking Add Application from the welcome menu.

Layout Template

Click on *Layout Template* from the Welcome Menu.

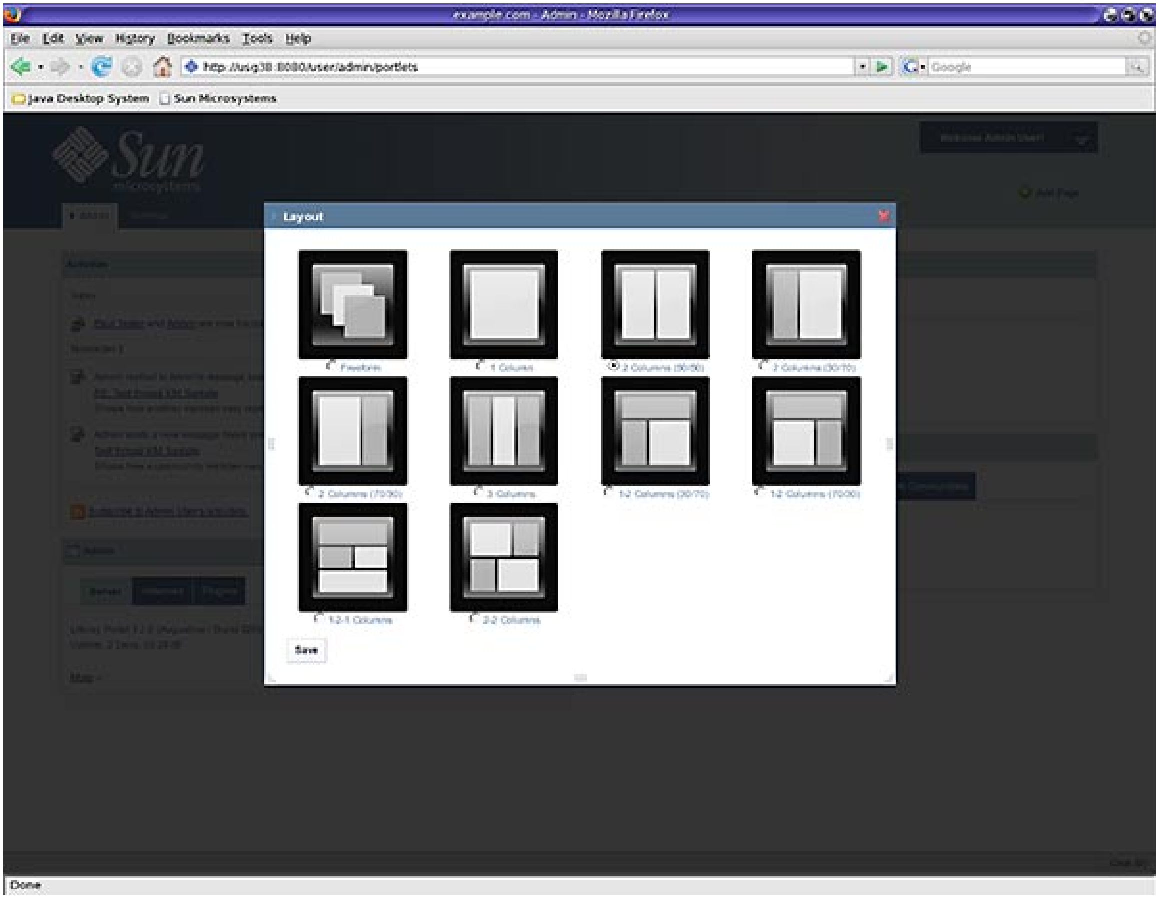


FIGURE 1–5 Layout Template

The *Layout* window appears. The *2 Column (50/50)* is the default layout. You can select the radio button for the Layout of your choice and click *Save* to apply the layout for your account.

Manage Pages

You can use Manage Pages from the Welcome Menu to specify a Friendly URL and apply other settings to a Public Page or a Private Page associated with your account.

You can refer the section on Friendly URLs.

Toggle Edit Controls

You can toggle edit controls off and on by clicking on *Toggle Edit Controls* from the Welcome Menu.

You can notice that edit controls are on when Toggle Edit Controls is on.

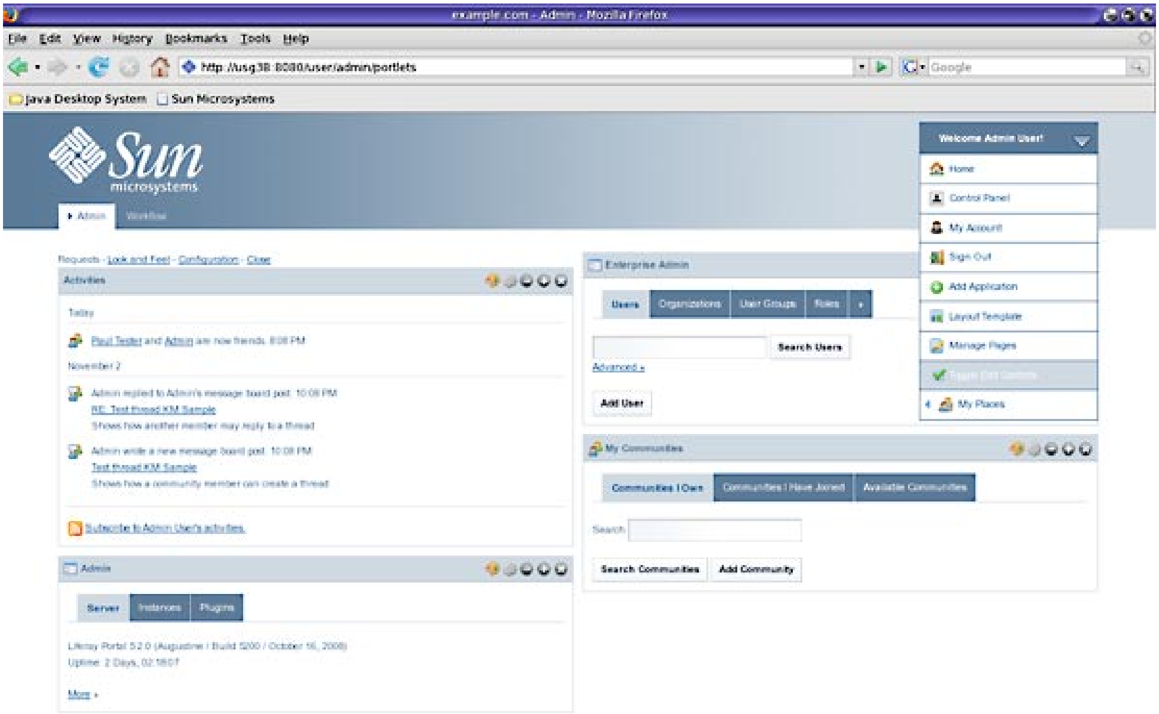
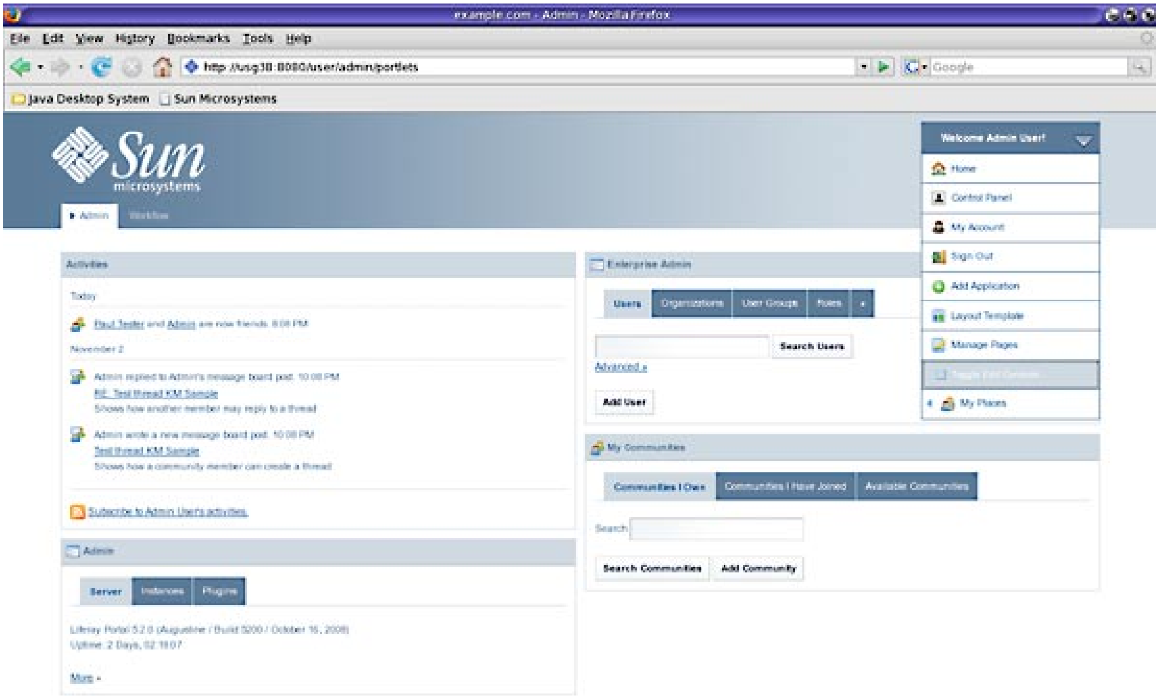


FIGURE 1–6 Toggle Edit Controls

The edit controls for the portlets on your page are off when 'Toggle Edit Controls' is off.



Today

Paul Tester and Admin are now friends. 8:08 PM

November 2

Admin replied to Admin's message board post. 10:08 PM
RE: Test thread KM Sample
Shows how another member may reply to a thread

Admin wrote a new message board post. 10:08 PM
Test thread KM Sample
Shows how a community member can create a thread

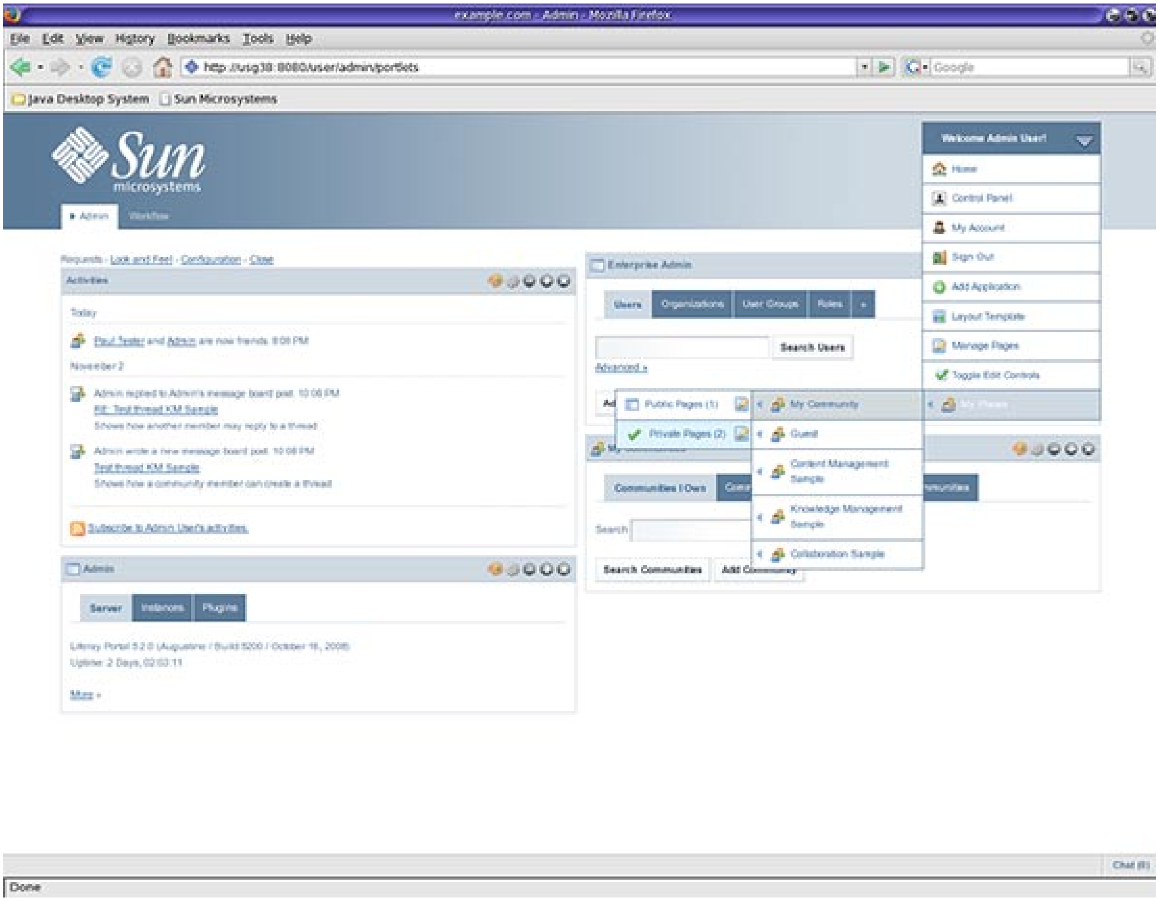
Subscribe to Admin User's activities.

Liferay Portal 5.2.0 (Augustine / Build 5200 / October 16, 2008)
Uptime: 2 Days, 02:18:07

FIGURE 1–7 Toggle Edit Controls

My Places

You can select *My Places* from the welcome menu to navigate to all the public and private pages of the communities associated with your account. When you click on My Places, the page you are on is highlighted in the menu appearing.



Today

Paul Tester and Admin are now friends. 8:09 PM

November 2

Admin replied to Admin's message board post. 10:08 PM
RE: Test thread KM Sample
Shows how another member may reply to a thread

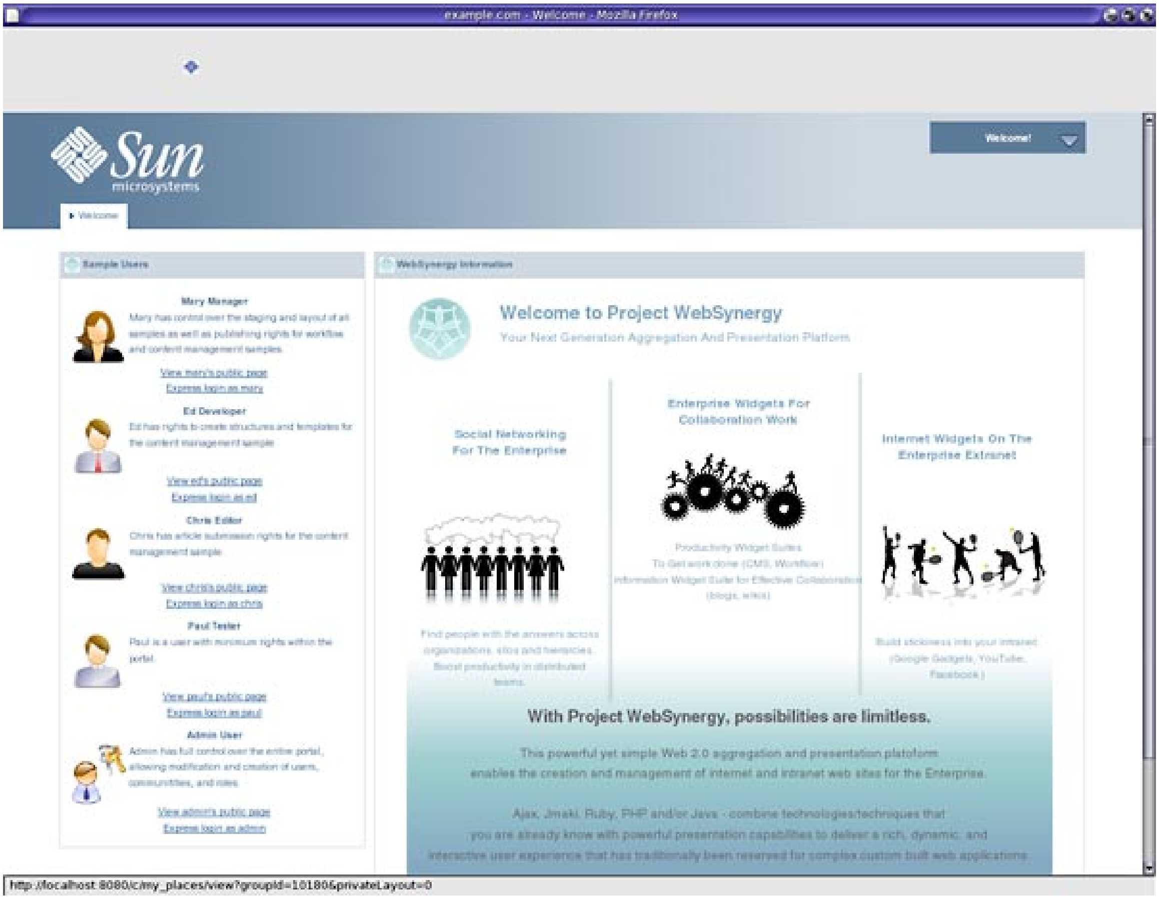
Admin wrote a new message board post. 10:08 PM
Test thread KM Sample
Shows how a community member can create a thread

Subscribe to Admin User's activities.

Liferay Portal 5.2.0 (Augustine / Build 5200 / October 16, 2008)
Uptime: 2 Days, 02:03:11

FIGURE 1–8 My Places menu

Note – A *public page*is a page which can be viewed by the Guest user i.e., the public page can be viewed without the need to login. *Private pages* can be viewed only after you login. For example, to view the public page of the admin user, click on the *View admin's public page* link. To view admin's private pages, you need to login as the admin user.



Mary Manager
Mary has control over the staging and layout of all samples as well as publishing rights for workflow and content management samples.
View mary's public page
Express login as mary

Ed Developer
Ed has rights to create structures and templates for the content management sample
View ed's public page
Express login as ed

Chris Editor
Chris has article submission rights for the content management sample
View chris's public page
Express login as chris

Paul Tester
Paul is a user with minimum rights within the portal.
View paul's public page
Express login as paul

Admin User
Admin has full control over the entire portal, allowing modification and creation of users, communitities, and roles
View admin's public page
Express login as admin

Welcome to Project WebSynergy
Your Next Generation Aggregation And Presentation Platform

Social Networking For The Enterprise
Find people with the answers across organizations, sites and hierarcies. Boost productivity in distributed teams.

Enterprise Widgets For Collaboration Work
Productivity Widget Suites
To Get work done (CMS, Workflow)
Information Widget Suite for Effective Collaboration
(blogs, wikis)

Internet Widgets On The Enterprise Extranet
Build stickiness into your intranet (Google Gadgets, YouTube, Facebook)

With Project WebSynergy, possibilities are limitless.

This powerful yet simple Web 2.0 aggregation and presentation platoform
enables the creation and management of internet and intranet web sites for the Enterprise.

Ajax, Jmaki, Ruby, PHP and/or Java - combine technologies/techniques that
you are already know with powerful presentation capabilities to deliver a rich, dynamic, and
interactive user experience that has traditionally been reserved for complex custom built web applications

http://localhost:8080/c/my_places/view?groupId=10180&privateLayout=0

FIGURE 1–9 My Places menu

Setting a Development Environment for WebSynergy

If you are beginning a WebSynergy-based project, chances are you will need to get your developers up and running before your production systems are ready. In order for a developer to do his or her work, an instance of WebSynergy needs to be running on his or her machine. Additionally, to prevent file-locking issues, a developer's version of WebSynergy should not use the embedded database, so a separate database will need to be installed.

WebSynergy makes use of many open source tools for development. This has two benefits: 1) It removes any barriers to entry, as there are no expensive tools to purchase in order to make use of WebSynergy, and 2) It allows WebSynergy to remain as tool-agnostic as possible. If developers wish to use an IDE to work on WebSynergy, great. If developers want to use a text editor and the command line, that's great too. Developers can choose the tools they are most comfortable with to write code on WebSynergy platform.

There are, however, some tools that are required in order to develop with WebSynergy. These are at a minimum:

- Apache Ant 1.7.0 or above
- A Java Development Kit
- A Liferay-supported database (MySQL recommended for a developer machine)
- The IDE or development environment of your choice

If you will be customizing WebSynergy via the Extension Environment (please see the WebSynergy Developer's Guide for further details), you may need:

- A Subversion client (optional: you can also download the Liferay source from the web site)

Developing With Apache Ant 1.7.0 or Above

Apache Ant is a build tool that is used extensively by both the Liferay source and the extension environment. You can download the latest version of Ant from `http://ant.apache.org`. After downloading, uncompress the archive into a folder. You then need to set the `ANT_HOME` environment variable to add binaries to your `PATH`.

▼ To Set the `ANT_HOME` Variable on Windows

1 **Go to Start -> Control Panel, and double-click the System icon.**

2 **Go to Advanced, and then click the Environment Variables button.**

3 **Under System Variables, select New to create the `ANT_HOME` environment variable.**

4 **Set the variable name to `ANT_HOME` and the variable value to *apache-install-dir*`\apache-ant-1.7.0` and click OK.**

For example, the installation directory might be `C:\Java\apache-ant-1.7.0`.

5 **Select New again to create the `ANT_OPTS` environment variable.**

6 **Set the variable name to `ANT_OPTS` and the variable value to `-Xms256M -Xmx512M` and click OK.**

7 **Select the `PATH` environment variable and select Edit.**

8 **Add `%ANT_HOME%\bin` to the end or beginning of the `PATH`.**

9 **Click OK and then click OK again.**

10 **To test the setting, type `ant` at a command prompt and press Return.**

- **If a `Build not found` error message appears, you have correctly installed Ant.**
- **If you do not see an error message, check the environment variable settings and make sure they are pointing to the correct directory locations.**

▼ To Set the `ANT_HOME` Variable on a Linux or Macintosh System

1 **In your home folder, open the `.bash_profile` file in the root directory.**

2 **Add the following lines to the file, substituting the directory where you installed Ant:**

```
ANT_HOME=/apache-install-dir/apache-ant-1.7.0
ANT_OPTS="-Xms256M -Xmx512M"
PATH=$PATH:$HOME/bin:$ANT_HOME/bin
export ANT_HOME ANT_OPTS PATH
```

3 **To test the setting, type `ant` at a command prompt and press Return.**

- **If a `Build not found` error message appears, you have correctly installed Ant.**
- **If you do not see an error message, check the environment variable settings and make sure they are pointing to the correct directory locations.**

CHAPTER 2

Installation

The WebSynergy evaluation bundle by default includes the HSQL database. HSQL, MySQL, Microsoft SQL, and Oracle 10g are the databases supported by WebSynergy.

You can use the evaluation bundle which has the HSQL database embedded with it for quick evaluation. It is a good method to have it up and running fast for reviewing or developing, it has several drawbacks:

- Only one user can access it at a time. This is because the data is stored on a file on disk and HSQL locks it when doing changes.
- The data is stored inside the application server and might be lost on redeployment.
- This configuration does not scale well and will have performance problems when multiple users are accessing the system.

Download WebSynergy (standalone). The standalone version which is not bundled with GlassFish, samples, and the HSQL database looks similar to `websynergy-for-gfv2.zip`. You can configure it to a database installed on your disk. MySQL, Microsoft SQL, and Oracle 10g are the databases supported by WebSynergy. We recommend that if your organization doesn't have a standard for local development databases, you should use MySQL for this, as it is small, free, and very fast.

WebSynergy is best suited to work on GlassFish v2, but it is also supported on other versions of GlassFish and also on other application servers.

Installing the Database, Application Server, and WebSynergy

Follow the below procedure to configure WebSynergy on Solaris with UTF-8 support. For the instructions to install MySQL on various other operating systems, see `http://dev.mysql.com/doc/refman/6.0/en/installing.html`.

▼ To Install and Configure MySQL

1. **Install MySQL by typing the following commands as superuser.**

   ```
   # groupadd mysql
   # useradd -g mysql mysql
   # cd /usr/local
   # gunzip install-dir/mysql-VERSION-OS.tar.gz | tar xvf -
   # ln -s install-dir/mysql-VERSION-OS mysql
   # cd mysql
   # chown -R mysql .
   # chgrp -R mysql .
   # scripts/mysql_install_db --user=mysql
   # chown -R root .
   # chown -R mysql data
   # bin/mysqld_safe --user=mysql & *
   ```

2. **Log in to the MySQL installation by going to the MySQL installation directory and typing the following command:**

   ```
   mysql -u root
   ```

3. **Create the UTF-8 database by typing the following commands:**

   ```
   mysql> create database lportal default character set utf8
   mysql> use lportal;
   mysql> create user lportal;
   mysql> grant all privileges on *.* to 'lportal'@'localhost' identified by 'lportal'
   mysql> set password for 'lportal'@'localhost' = password('lportal');
   ```

4. **Enable UTF-8 support for MySQL.**

   See, “To Enable UTF-8 Support for MySQL” on page 38

▼ To Install GlassFish

Installing and starting GlassFish is a prerequisite for installing WebSynergy.

1. **Download the GlassFish v2 distribution.**

   `https://glassfish.dev.java.net/public/downloadsindex.html`.

2. **Set the environment variable JAVA_HOME to point to the directory in which JDK 1.6 is installed.**

3. **Run the following command:**

   ```
   java -Xmx256m -jar <filename>.jar
   ```

4 **Set ANT_HOME to point to the installation of ant 1.7 or above, which is at** *<downloaded-dir>*`/glassfish/lib/ant` directory.

5 **Navigate to the GlassFish directory:**

```
cd glassfish
```

6 **Run the following command:**

```
ant -f setup.xml
```

7 **Start GlassFish.**

```
cd bin
```

<downloaded-dir>`/glassfish/bin>/asadmin start-domain`

To Install WebSynergy

1 **Download the standalone version of WebSynergy.**

2 **Extract the WebSynergy bundle.**

```
unzip websynergy-for-gfv2.zip
```

3 **Navigate to the Application folder.**

```
cd websynergy/application
```

4 **Run** `ant -f install.xml`**.**

```
ant -f install.xml
Buildfile: install.xml

show-user-warning:
   [input] Ant must be 1.7 or greater. JAVA_HOME must be set to JDK 1.5 or greater.
Glassfish must be running. [RETURN to continue or CONTROL-C to stop]


set-glassfish-properties:
   [input] Enter Glassfish Directory  [/opt/glassfish]
/424/glassfish
   [input] Enter Glassfish Domain (include full path to domain)  [/424/glassfish/domains/domain1]

   [input] Enter Glassfish Target  [server]

   [input] Enter Glassfish Administrator Password  [adminadmin]

   [input] Enter Glassfish Administration Port  [4848]
   [input] Is Glassfish Clustered [Yes/No]  [No]


set-database-properties:
   [input] Use builtin HSQL or mySQL database  [HSQL]
mySQL

set-hsql-properties:

set-mssql-properties:

set-mysql-properties:
   [input] Enter Database User Name  [root]

   [input] Enter Database User Password  []
password
   [input] Enter Database Host  [localhost]

   [input] Enter Database Port  [3306]

   [input] Enter Database Name  [lportal]
```

You need to set the application server and database properties in the process of running `install.xml`. It is also required that Ant must be 1.7 or greater and JDK must be 1.5 or greater (JDK 1.6 is preferred). Also, GlassFish must be running.

▼ To Enable UTF-8 Support for MySQL

You need to make changes to the `portal-ext.properties` file to enable `UTF-8` support for MySQL. The WebSynergy evaluation bundle has a `portal-ext.properties` file in the *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`. When you are using a WebSynergy bundle which is not including samples, you have to create a `portal-ext.properties` file. This task describes the process to create a `portal-ext.properties` file, and making changes to it to enable `UTF-8` support for MySQL.

1 **Create a temporary folder. In this example, name the folder as test.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
mkdir /tmp/test
```

2 **Copy the `portal-impl.jar` file to the temp folder.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
cp GlassFish
install-dir/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/portal-impl.
/tmp/test
```

3 **Change directory to `temp`.**

Equivalent step in the command prompt or in the terminal window of SOLARIS or any UNIX like OS:

```
cd /tmp/test
```

4 **Extract the `portal-impl.jar` file.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
jar -xvf portal-impl.jar
```

After `unjar` it will have `portal.properties` file in the same directory.

5 **Open `portal.properties` and copy the "`JDBC`" property.**

6 **Create a text file (using any text editor) and name it as `portal-ext.properties`, and paste the "`JDBC`" property into it.**

```
#
    # Hypersonic
    #
    jdbc.default.driverClassName=org.hsqldb.jdbcDriver
    jdbc.default.url=jdbc:hsqldb:lportal
    jdbc.default.username=sa
    jdbc.default.password=

    #
    # MySQL
    #
    #jdbc.default.driverClassName=com.mysql.jdbc.Driver
    #jdbc.default.url=jdbc:mysql://localhost/lportal?useUnicode=true&characterEncoding=
UTF-8&useFastDateParsing=false
    #jdbc.default.username=
    #jdbc.default.password=
```

7 **Comment Hypersonic (HSQL) and uncomment MySQL.**

8 **Save the `portal-ext.properties` file.**

9 **Copy the `portal-ext.properties` file and paste it to** *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`.

10 **Restart GlassFish server.**

CHAPTER 3

Customizing Project WebSynergy

This chapter describes some simple customization tasks; most of which can be done from the user interface of WebSynergy. For accomplishing more serious changes to the product configuration, you need to make changes to the `portal-ext.properties` file. A simple example for making changes to the `portal-ext.properties` file, see "Localization" on page 56. For details of properties in the `portal-ext.properties` file, refer "The *portal-ext.properties* File" on page 83.

- Branding and Site Identity
- Friendly URLs
- Site Layout and Navigation
- Using Bundled WebSynergy Portlets
- "Customizing the Look and Feel and Configuration of a Portlet" on page 52
- "Localization" on page 56

Branding and Site Identity

To be done.

- "To change the Banner Logo" on page 41
- "To change the Address Bar and Browser tab Icons" on page 43

▼ To change the Banner Logo

1 **Login to Project WebSynergy as admin user.**

2 **Click Add Applications and expand the Admin folder.**

3 **Click Add against the Enterprise Admin portlet.**

The *Enterprise Admin* portlet is added to your page

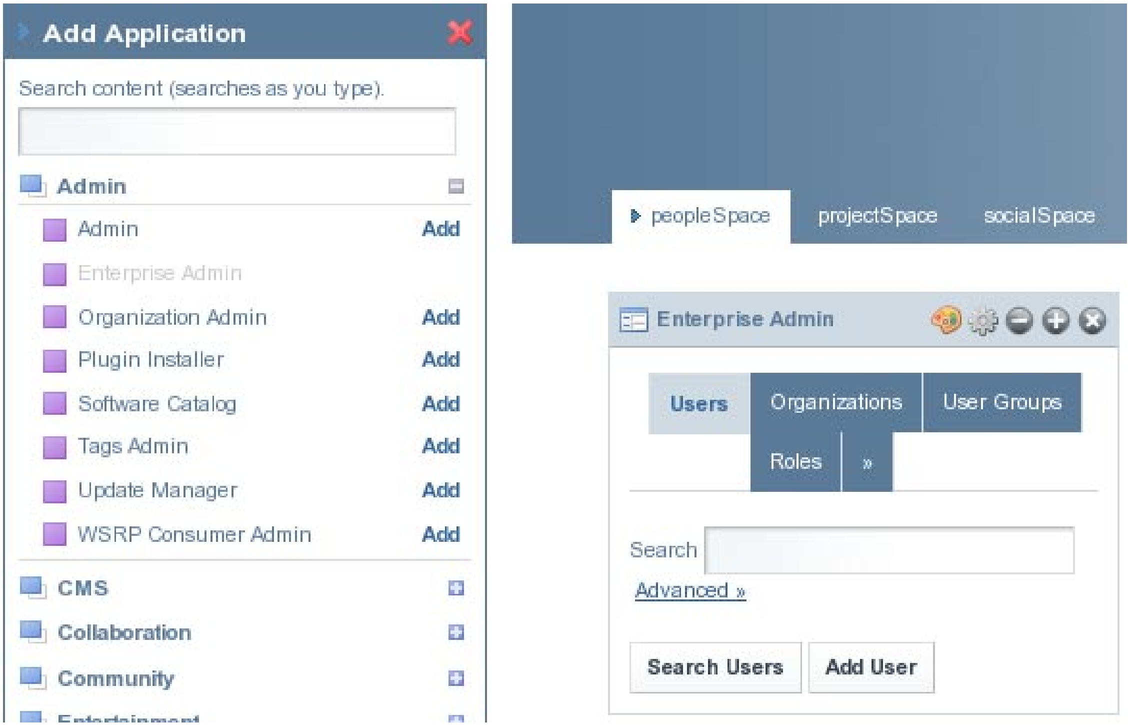
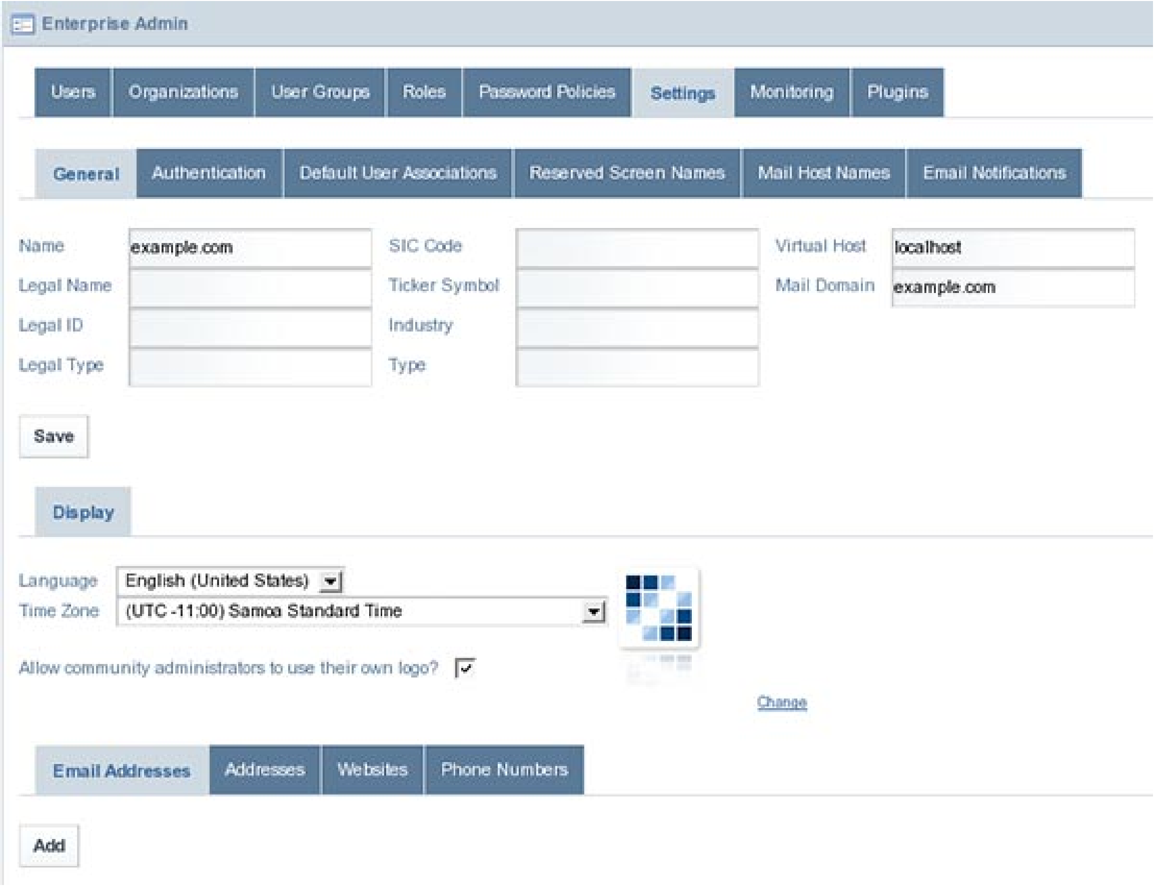


FIGURE 3–1 Enterprise Admin portlet

4 **Maximize the portlet, and click on the Settings tab.**



Name: example.com

Virtual Host: localhost

Mail Domain: example.com

Language: English (United States)

Time Zone: (UTC -11:00) Samoa Standard Time

Allow community administrators to use their own logo? ☑

FIGURE 3–2 Settings page

5 **Click on the Change link. You are prompted to browse and select a new banner logo.**

▼ To change the Address Bar and Browser tab Icons

1 **Navigate to the directory which contains the WebSynergy installation.**

2 **From the directory containing the WebSynergy installation, navigate to** `/glassfish/domains/domain1/applications/j2ee-modules/websynergy/html/themes/classic/i`

3 **Change the liferay.ico image.**

4 **Clear the browser cache and reload the page.**

Friendly URLs

To be done.

▼ To change the Friendly URL of a Page

1. **Login to Project WebSynergy as admin user.**

2. **Click on Manage Pages from the welcome menu.**

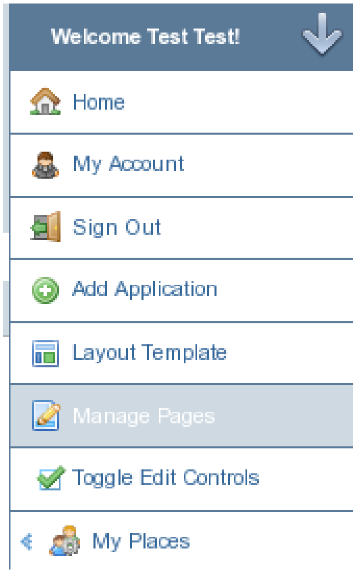
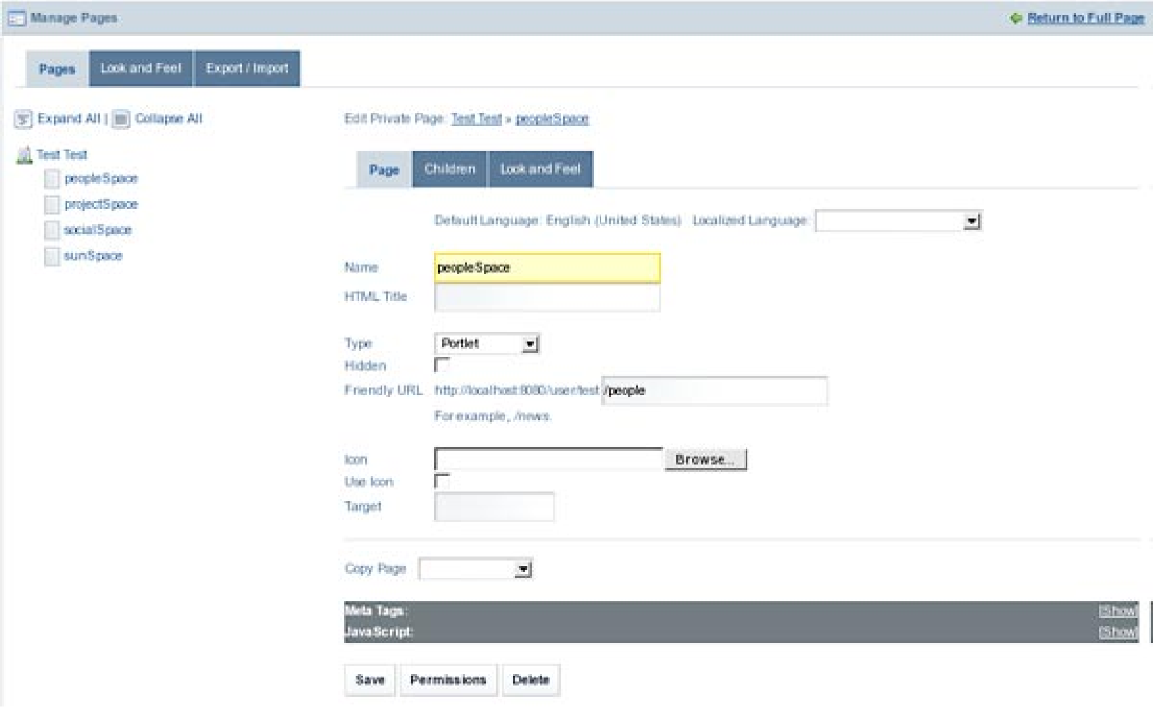


FIGURE 3–3 Welcome menu – Manage Pages

You can see all the pages on the left hand side under the Guest (Test Test).



Test Test
- peopleSpace
- projectSpace
- socialSpace
- sunSpace

Edit Private Page: Test Test » peopleSpace

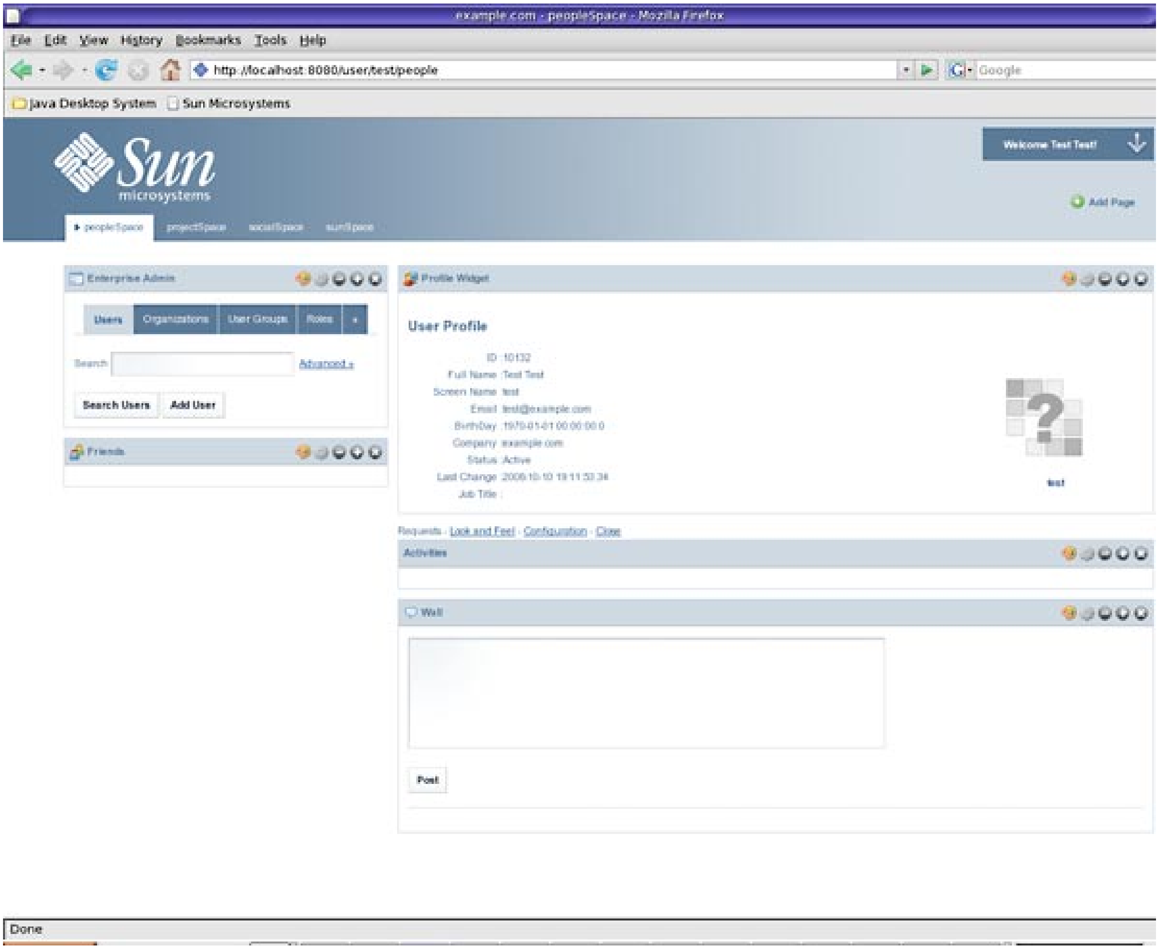
Default Language: English (United States) Localized Language:

Name	peopleSpace
HTML Title	
Type	Portlet
Hidden	□
Friendly URL	http://localhost:8080/user/test /people
	For example, /news.
Icon	
Use Icon	□
Target	
Copy Page	

Meta Tags: Show
JavaScript: Show

FIGURE 3–4 Manage Pages page

Friendly URL for the *PeopleSpace* page is `http://localhost:8080/user/test/people`. This means, you can directly access the PeopleSpace page by typing the URL `http://localhost:8080/user/test/people`, as seen in the below figure.



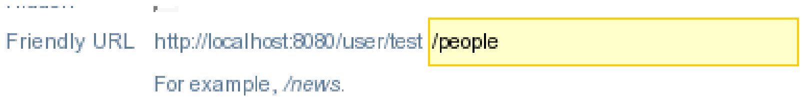
User Profile

ID :10132
Full Name :Test Test
Screen Name :test
Email :test@example.com
BirthDay :1970-01-01 00:00:00.0
Company :example.com
Status :Active
Last Change :2008-10-10 19:11:53.34
Job Title :

test

FIGURE 3–5 Friendly URL for the page

You can note the at value in the Friendly URL field looks as below:



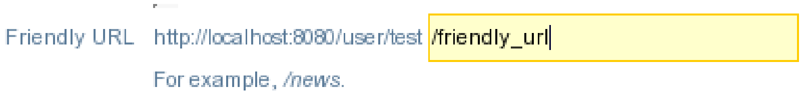
Friendly URL http://localhost:8080/user/test /people

For example, /news.

FIGURE 3–6 Friendly URL field

3 **Change the value of the Friendly URL to the value you like.**

In our example, we change it from *people* to *friendly_url*.



Friendly URL http://localhost:8080/user/test /friendly_url

For example, /news.

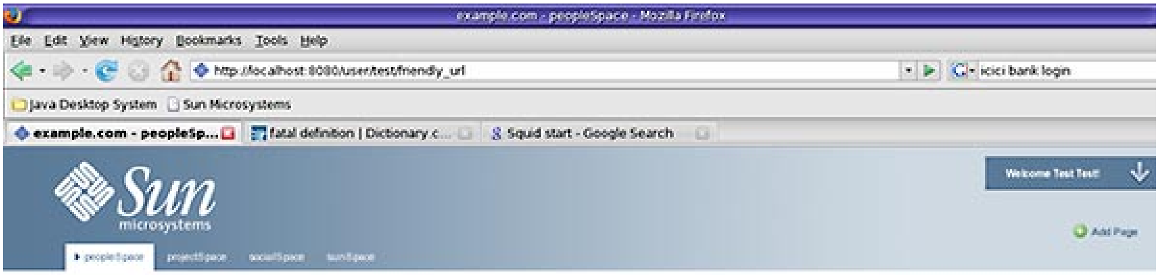
FIGURE 3–7 Friendly URL field with new Friendly URL

4 **Click Save.**

A message saying that the page is not found appears. For the changed URL to take effect, logout and login back to WebSynergy.

5 **Now you can view the Friendly URL.**

The changed Friendly URL for the PeopleSpace page is `http://localhost:8080/user/test/friendly_url`.



http://localhost:8080/user/test/friendly_url

example.com - peopleSpace - Mozilla Firefox
File Edit View History Bookmarks Tools Help
icici bank login
Java Desktop System
Sun Microsystems
example.com - peopleSp...
fatal definition | Dictionary.c...
Squid start - Google Search
Sun microsystems
Welcome Test Test!
Add Page
peopleSpace
projectSpace
socialSpace
sunSpace

FIGURE 3–8 New Friendly URL

Site Layout and Navigation

Layout Templates are ways of choosing how your portlets will be arranged on a page. They make up the body of your page, the large area where you drag and drop your portlets to create your pages. WebSynergy comes with several built-in layout templates, but if you have a complex page layout (especially for your home page), you may wish to create a custom layout template of your own. You can refer to the development document in Liferay Documentation for the procedure to create a custom Layout Template.

- “To change the Site Layout” on page 48
- “To create a new Layout” on page 49
- “Changing the Page Theme” on page 49

▼ To change the Site Layout

1. **Login to Project WebSynergy as Admin user.**

2. **Click on Layout Template from the Welcome menu.**

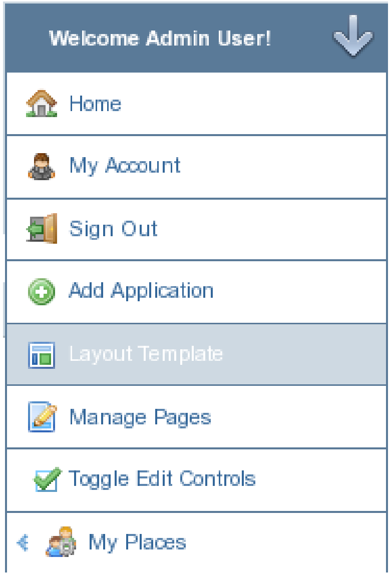
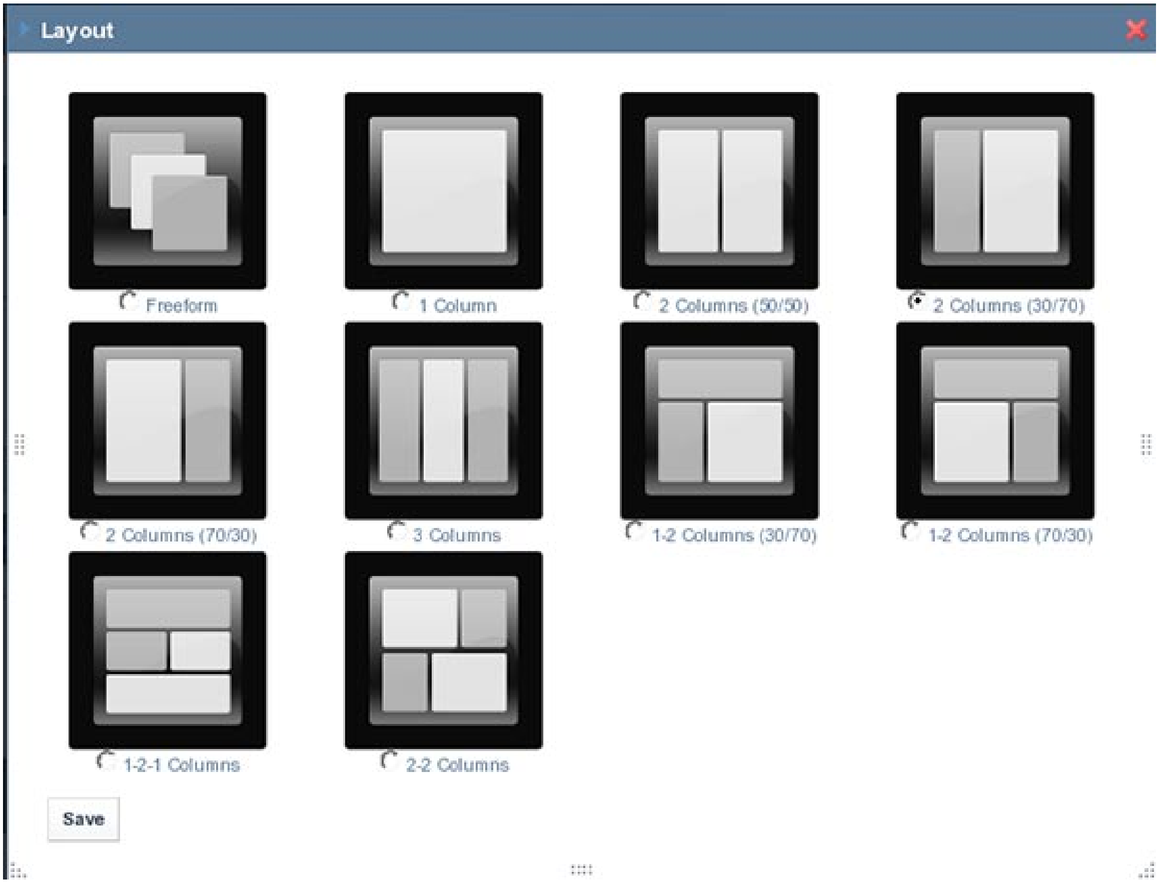


FIGURE 3–9 Welcome menu – Layout Template

3. **The Layout page appears.**



Layout

○ Freeform
○ 1 Column
○ 2 Columns (50/50)
● 2 Columns (30/70)
○ 2 Columns (70/30)
○ 3 Columns
○ 1-2 Columns (30/70)
○ 1-2 Columns (70/30)
○ 1-2-1 Columns
○ 2-2 Columns

FIGURE 3–10 Layout Templates

4 **Select the Layout you want to use and click Save.**

▼ To create a new Layout

- **To be done.**

Changing the Page Theme

Themes are hot deployable plugins which can completely transform the look and feel of the portal. Most organizations have their own look and feel standards which go across all of the web sites and web applications in the infrastructure. WebSynergy makes it possible for a site designer to create a theme plugin which can then be installed, allowing for the complete transformation of the portal to whatever look and feel is needed. You can refer to the development document in Liferay Documentation for the procedure to create a custom theme.

▼ To Change the Page Theme

1. **Login to Project WebSynergy.**

   Any user can change themes, but only the admin user has access to all available themes, and has the privilege to install more themes.

2. **Choose Manage Pages from the welcome menu.**

3. **Click on the Look and Feel tab.**

   The Current Theme for the page and all the available themes are displayed.

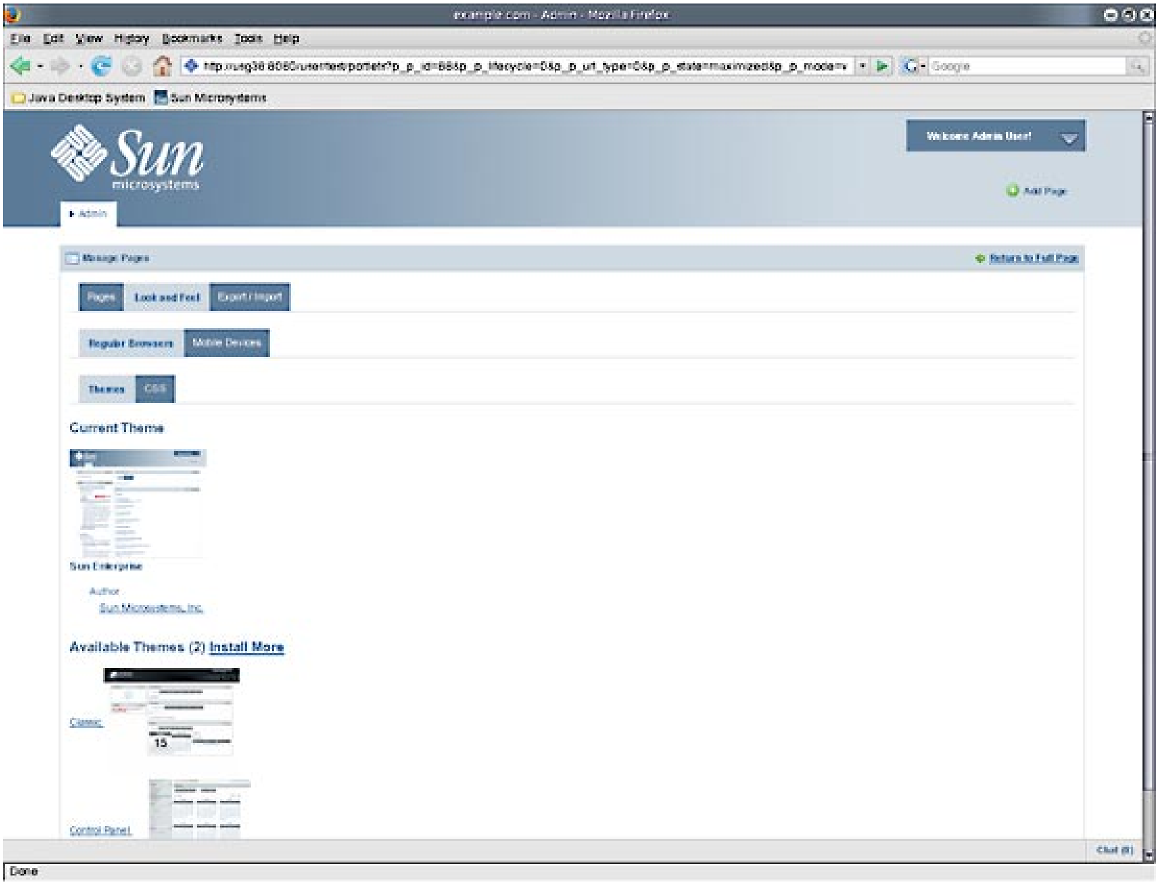


FIGURE 3–11 Page Theme

4 **To change the theme, click on a theme from Available Themes.**

The theme is applied to your page.

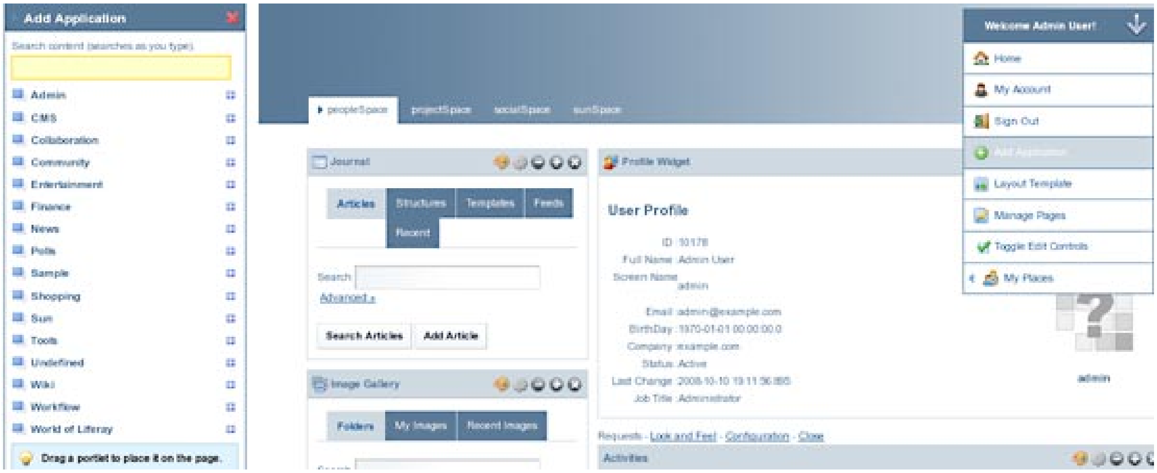
Using Bundled WebSynergy Portlets

To be done.

▼ To use Bundled WebSynergy Portlets

1 **Login to Project WebSynergy as Admin user.**

2 **Click on Add Applications from the welcome menu.**



User Profile

ID: 10178
Full Name: Admin User
Screen Name: admin
Email: admin@example.com
BirthDay: 1970-01-01 00:00:00.0
Company: example.com
Status: Active
Last Change: 2008-10-10 19:11:56.895
Job Title: Administrator

FIGURE 3–12 Bundled WebSynergy Portlets

3 **Add the required portlet to your page.**

Customizing the Look and Feel and Configuration of a Portlet

You can change the display name of a portlet and also its look and feel.

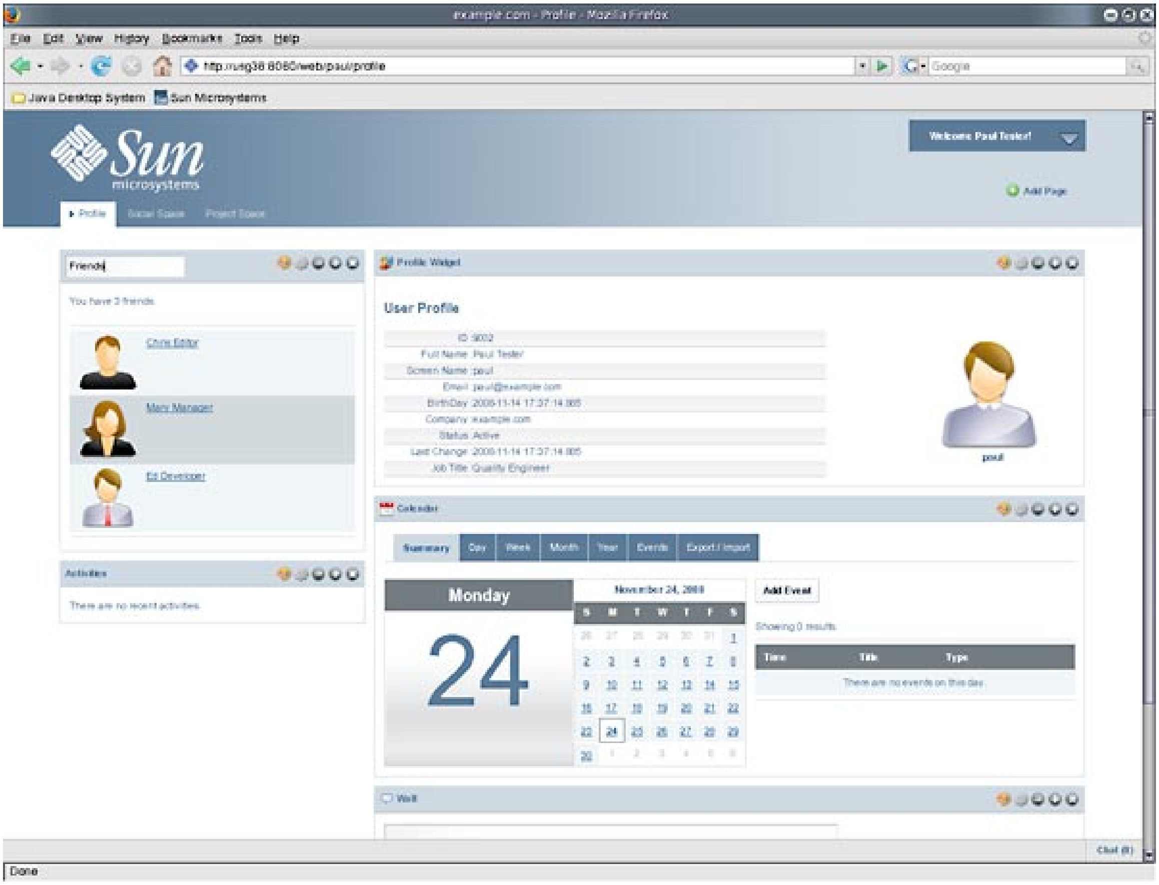
Changing the Name of a Portlet

You can change the display name of a portlet simply by clicking on the name of the portlet.

▼ To Change the Name of a Portlet which is on a Private Page

1 **Login to Project WebSynergy as any user of your choice.**
In this example, login as Paul Tester.

2 **Click on the name of a portlet.**
For example, click on the name of the Friends portlet.



You have 3 friends.

There are no recent activities.

ID: 9002
Full Name: Paul Tester
Screen Name: paul
Email: paul@example.com
BirthDay: 2008-11-14 17:37:14.805
Company: example.com
Status: Active
Last Change: 2008-11-14 17:37:14.805
Job Title: Quality Engineer

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

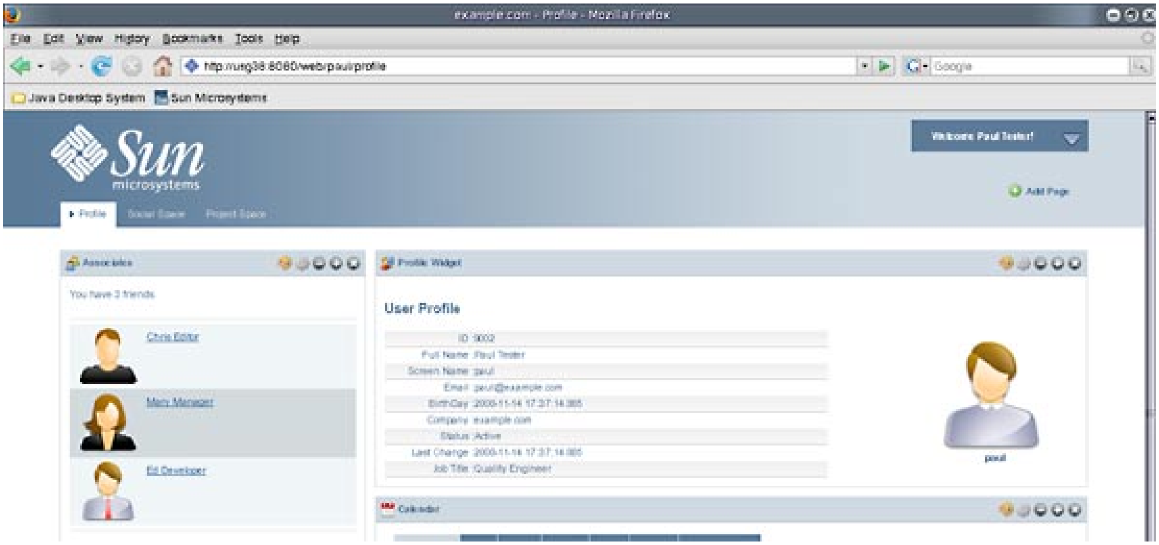
Showing 0 results.

Time	Title	Type
There are no events on this day.		

FIGURE 3–13 Changing the display name of a portlet

3 Edit the display name.

In this example change the name of the portlet to *Associates*.



You have 2 friends

Chris Editor

Mary Manager

Ed Developer

User Profile

ID 9002
Full Name Paul Tester
Screen Name paul
Email paul@example.com
BirthDay 2008-11-14 17:37:14.985
Company example.com
Status Active
Last Change 2008-11-14 17:37:14.985
Job Title Quality Engineer

FIGURE 3–14 Changed display name

To Change the Name of a Portlet which is on a Public Page

To change the name of a portlet which is on the public page of a user, you need to login as the user and navigate to the public page of the user from *My Places* on the welcome menu.

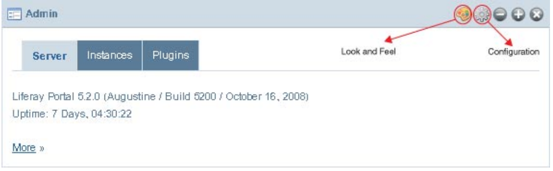
1. **Login to Project WebSynergy as any user of your choice.**

2. **Navigate to a public page of the user by selecting My Places on the welcome menu.**

3. **Click on the name of a portlet, and edit and change the name.**

Customizing the Look and Feel and Configuration of a Portlet

The Look and Feel and Configuration buttons on a portlet can be used to customize the portlet.



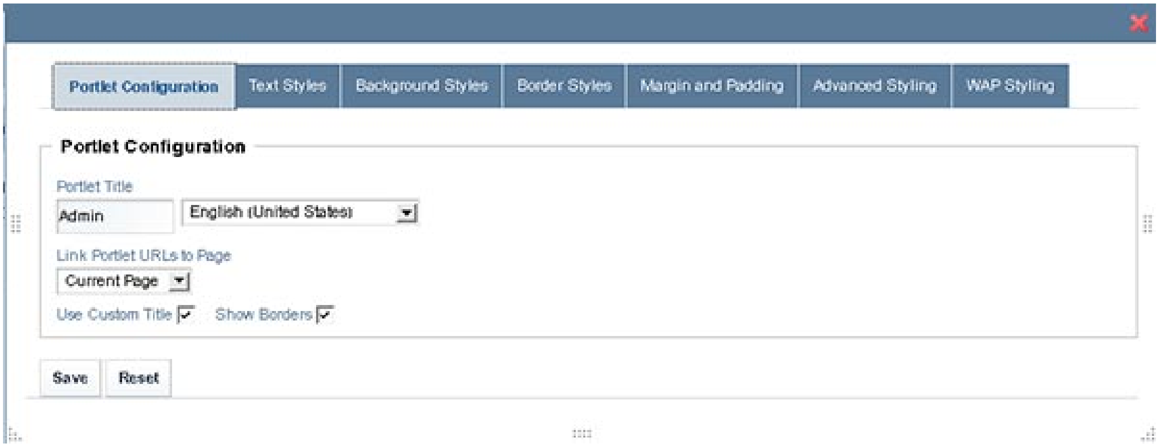
Liferay Portal 5.2.0 (Augustine / Build 5200 / October 16, 2008)
Uptime: 7 Days, 04:30:22

FIGURE 3–15 Look and Feel and Configuration buttons on a portlet

- "The Look and Feel Button" on page 55
- "The Configuration Button" on page 56

The Look and Feel Button

A window pops-up on clicking the Look and Feel button. The window provides options to customize the look and feel of the portlet.



Portlet Configuration

Portlet Title: Admin — English (United States)

Link Portlet URLs to Page: Current Page

Use Custom Title ☑ Show Borders ☑

FIGURE 3–16 Customizing the Look and Feel

The window has tabs for Portlet Configuration, Text Styles, Background Styles, Border Styles, Margin and Padding, Advanced Styling, and WAP Styling.

Portlet Configuration tab

If you enable the *Use Custom Title* option, you can change the value in the *Portlet Title* field. When you click Save, the new title for the portlet takes effect. The *Show Borders* option can be used to enable or disable the display of borders.

Text Styles tab

Using this tab, you can change the default settings for text such as, font, size, color, alignment, bold, and italic.

Background Styles tab

Using this tab, you can change the default background color for the portlet.

Border Styles tab

Using this tab, you can change the border styles for the portlet. You can change the default settings for the Border Width, Border Style, and Border Color. You can set different width, style, and colors for different borders by disabling the *Same for All* option.

Margin and Padding tab

Using this tab, you can change the default margin and padding settings. It may not be generally advisable to change the default margin and padding settings.

Advanced Styling tab

Using this tab, you can define CSS rules for the portlet and all similar portlets.

WAP Styling tab

Using this tab, you can define the styling for mobile handsets.

The Configuration Button

A window pops-up on clicking the Customization button. From this window, you can configure permissions for the portlet, can export/import the portlet data, and can enable sharing on the portlet/widget.

For a detailed use case on the Configuration button to enable sharing of portlets, see “Enabling Sharing of a Portlet” on page 258

Localization

WebSynergy is localized out-of-box in 22 languages , out of which Sun provides localization support for Japanese (`ja`) and Simplified Chinese (`zh_CN`). Remaining languages are community driven.

- “Setting User Prefered Language in WebSynergy” on page 57

Setting User Prefered Language in WebSynergy

User can use WebSynergy in their preferred language from any of the languages supported by it. This section describes how users can change their preferred language.

▼ To set User Language

1 **Login to WebSynergy as admin user.**

2 **Click 'My Account' from the welcome dock, it takes you to the Control Panel.**

3 **Click on the 'Display Settings' link under Miscellaneous in the right menu.**

4 **Select a language from the Language drop down.**

5 **Click Save.**

User interface changes to the language selected. You can repeat the above steps to change the language again in the feature.

Note – You can use the same page to change the time zone and the greeting message.

Adding a new Language Support

You can add localization support for a language which is not supported by WebSynergy by default. This section provides details on how you can add localization support for a new language.

▼ To add new Language Support

The WebSynergy evaluation bundle has a `portal-ext.properties` file in the *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`. When you are using a WebSynergy bundle which is not including samples, you have to create a `portal-ext.properties` file. This task describes the process to create a `portal-ext.properties` file, and making changes to it to support localization.

1. **Create a temporary folder. In this example, name the folder as test.**

   Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

   ```
   mkdir /tmp/test
   ```

2. **Copy `portal-impl.jar` to the temp folder.**

   `portal-impl.jar` contains language resource bundles for WebSynergy. This `jar` file is located under the *GlassFish install-dir*`/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/` directory.

   Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

   ```
   cp GlassFish
   install-dir/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/portal-impl.jar
   /tmp/test
   ```

3. **Change directory to `temp`.**

   Equivalent step in the command prompt or in the terminal window of SOLARIS or any UNIX like OS:

   ```
   cd /tmp/test
   ```

4. **Extract the `portal-impl.jar` file.**

   Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

   ```
   jar -xvf portal-impl.jar
   ```

   After `unjar` it will have `portal.properties` file in the same directory.

5. **Open `portal.properties` and copy the `locales` property from the `Languages and Time Zones` section.**

6. **if `portal-ext.properties` does exits in WebSynergy then paste the `locales` property into it. Otherwise, create a text file (using any text editor) and name it as `portal-ext.properties`, and paste the `locales` property into it.**

   Default location of `portal-ext.properties` is *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes/portal-ext.p`

7. **Modify the `locales` to add the new locale at the end. Add new locale in the format of "LanguageCode_CountryCode".**

   See the following links to find language and country codes:

   - See, `http://ftp.ics.uci.edu/pub/ietf/http/related/iso639.txt` for language codes.
   - See, `http://userpage.chemie.fu-berlin.de/diverse/doc/ISO_3166.html` for country codes.

For example, if you are adding support for Hindi, you need to add `hi_IN` at the end of `locales` property as shown below:

```
locales=ar_SA,ca_AD,ca_ES,zh_CN,zh_TW,cs_CZ,nl_NL,en_US,fi_FI,fr_FR,de_DE,el_GR,hu_HU,
it_IT,ja_JP,ko_KR,nb_NO,fa_IR,pt_BR,ru_RU,es_ES,sv_SE,tr_TR,vi_VN,hi_IN
```

8 **Save the** `portal-ext.properties` **file.**

9 **Navigate to the** `content` **folder and open the** `Language.properties` **file.**

Note – A folder named `content` is created on extracting the `portal-impl.jar` file using the `unjar` command.

---

10 **Translate all the strings you want to localize from** `temp/test/content/Language.properties` **to the new language.**

11 **Rename the translated** `Langauge.properties` **file to** `Language-ext_<`*LanguageCode*`>.properties.native`**.**

12 **Run the following commands on the directory which contains the** `Language-ext_<`*LanguageCode*`>.properties.native` **file:**

```
native2ascii -encoding UTF-8 Language-ext_<LanguageCode>.properties.native
Language-ext_<LanguageCode>.properties
```

13 **Copy the** `portal-ext.properties` **file (under** `temp/test`**), and paste it to** *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`**.**

14 **Copy the** `Language-ext_<LanguageCode>.properties.native` **file (under** `temp/test/content`**) and the** `Language-ext_<LanguageCode>.properties` **files (under** `temp/test/content`**), and paste them to** *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes/content`

15 **Restart GlassFish server.**

To Verify the Localization of the Interface

1 **Log in to WebSynergy as admin user.**

2 **Click on My Account from the welcome dock, it takes you to the Control Panel.**

3 **Click on the Display Settings link under Miscellaneous.**

The new language setting is reflected in the Language drop-down menu.

4 **Select the language for the user and click Save.**

The user interface is changed to the new language.

Removing Localization Support for Unwanted Languages

Generally, users don't require localization support for all the languages which are supported by WebSynergy. This section describes how you can remove localization support for the languages which are not required.

▼ To Remove Localization Support for Unwanted Languages

Overwrite the `locales` property of `portal.properties` to remove unwanted languages. You need to make sure that the `portal-ext.properties` file is placed at *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`.

1 **Create a temporary folder. In this example, name the folder as test.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
mkdir /tmp/test
```

2 **Copy `portal-impl.jar` to the temp folder.**

`portal-impl.jar` contains language resource bundles for WebSynergy. This `jar` file is located under the *GlassFish install-dir*`/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/` directory.

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
cp GlassFish
install-dir/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/portal-impl.jar
/tmp/test
```

3 **Change directory to `temp`.**

Equivalent step in the command prompt or in the terminal window of SOLARIS or any UNIX like OS:

```
cd /tmp/test
```

4 **Extract the `portal-impl.jar` file.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
jar -xvf portal-impl.jar
```

After `unjar` it will have `portal.properties` file in the same directory.

5 **Open `portal.properties` and copy the `locales` property from the `Languages and Time Zones` section.**

6 **if `portal-ext.properties` does exits in WebSynergy then paste the `locales` property into it. Otherwise, create a text file (using any text editor) and name it as `portal-ext.properties`, and paste the `locales` property into it.**

Default location of `portal-ext.properties` is *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes/portal-`

7 **Edit the `locales` property to include only the languages which are needed.**

For example, If you want to support only English, German and Spanish, remove all the unwanted locales so that the `locales` value looks as below:

```
locales=en_US,de_DE,es_ES
```

8 **Save the `portal-ext.properties` file.**

9 **Copy the `portal-ext.properties` file (under `temp/test`), and paste it to *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`.**

10 **Restart the GlassFish server.**

Enabling the Guest Users to Select a Browser Language

Follow the below procedure to enable guest users to select their preferred browser language.

▼ To Enable Guest Users to Select a Browser Language

Overwrite `locale.default.request` property in the `Languages and Time Zones` section of `portal.properties` to enable guest users to select their preferred browser language. You need to set this property value to `true` and put it in `portal-ext.properties`.

1 **Create a temporary folder. In this example, name the folder as test.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
mkdir /tmp/test
```

2 **Copy `portal-impl.jar` to the temp folder.**

`portal-impl.jar` contains language resource bundles for WebSynergy. This `jar` file is located under the *GlassFish install-dir*`/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/` directory.

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

cp *GlassFish install-dir***/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/lib/portal-impl.jar /tmp/test**

3 **Change directory to `temp`.**

Equivalent step in the command prompt or in the terminal window of SOLARIS or any UNIX like OS:

```
cd /tmp/test
```

4 **Extract the `portal-impl.jar` file.**

Equivalent step in the command prompt or in the terminal window of a UNIX like OS:

```
jar -xvf portal-impl.jar
```

After `unjar` it will have `portal.properties` file in the same directory.

5 **Open `portal.properties` and copy the `locale.default.request` property from the `Languages and Time Zones` section.**

6 **if `portal-ext.properties` does exits in WebSynergy then paste the `locale.default.request` property into it. Otherwise, create a text file (using any text editor) and name it as `portal-ext.properties`, and paste the `locale.default.request` property into it.**

Default location of `portal-ext.properties` is *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes/portal-ext.p`

7 **Edit the property to set it to `locale.default.request=true`.**

8 **Save the `portal-ext.properties` file.**

9 **Copy the `portal-ext.properties` file (under `temp/test`), and paste it to** *GlassFish install-dir*`/domains/domain1/applications/j2ee-modules/websynergy/WEB-INF/classes`.

10 **Restart the GlassFish server.**

Note – Clear browser cookies before accessing WebSynergy interface to allow above changes to take place.

Hierarchy of Precedence for *Language.properties* files

Language/display information is defined in `Language.properties`, as well as various other `Language_`<*LanguageCode*>`.properties` files for foreign languages. These files are obtained on extracting the `portal-impl.jar` file, and are placed in the `content` folder. `Language.properties` is the primary and "default" language definition file, but definitions in that file may be overridden by language specific definitions. For example, the file `Language_en.properties` contains the English versions of most of the definitions. The file `Language_fr.properties` contains the French version, etc.

Language files can further have locale specific definitions. The `Language_en_US.properties` file (if it exists), contains the English phrase variations further defined for the United States. Other languages can also have locale specific definitions.

Rules of Precedence

The following is the hierarchy of precedence for different `Language.properties` files to override each other:

1. `Language-ext.properties` takes precedence over `Language.properties`.
2. Language specific versions take precedence over the non language specific versions. For example, `Language_en.properties` takes precedence over `Language.properties`.
3. Location specific versions take precedence over the non location specific versions. For example, `Language_en_US.properties` takes precedence over `Language_en.properties`.

For the `Language.properties` file for English, the following is the hierarchy of precedence, where `Language-ext_en_US.properties` will have the highest precedence and the `Language.properties` file will have the least precedence:

1. `Language-ext_en_US.properties`
2. `Language_en_US.properties`
3. `Language-ext_en.properties`
4. `Language_en.properties`
5. `Language-ext.properties`
6. `Language.properties`

Changing the Existing Language Entries

WebSynergy allow a easy way to change existing language entries. You can change English and foreign languages entries. Consider the example of changing the name of a portlet from *Message Boards* to *Forums*. To do this in the case when English is the default language, you need to create the `Language-ext_en.properties` to override the `Language_en.properties` file. Replicate `Language-ext_en.properties` with the content in the `Language_en.properties` file, and search and replace the entry `javax.portlet.title.19=Message Boards` with `javax.portlet.title.19=Forums`. The portlet *Message Boards* is renamed to *Forums*.

WebSynergy also allows you to add new language entries for customization or for new portlets. For example, you can make an entry similar to `javax.portlet.title.EXT_1=Reports`, to add the name of a new portlet named *Reports*.

Customizing the Name of a Portlet for a Language

Consider the example of localizing WebSynergy to French. In this case, name of the Admin portlet translates as `Administration du portail`. You may not find this appropriate, and may want to have a customized name for the Admin portlet when WebSynergy is localized to French. You can customize the name of a portlet for any language by using the *Look and Feel* button on a portlet.

▼ To Customize the Name of a Portlet for a Language

1. **Login the WebSynergy as admin user.**

2. **Click on the 'Look and Feel' button on a portlet.**

   In this example, click on the *Look and Feel* button on the *Admin* portlet.

3. **Select the 'Use Custom Title' option.**

4. **Select a language from the language list.**

   In this example, select French.

5. **Type a title for the portlet in the selected language in the 'Portlet Title' box.**

   In this example, specify the name of the portlet as `Administration`.

6. **Click Save.**

   Note – For the changes to take effect, you may have save twice with 'Use Custom Title' option canceled and selected respectively. Some times the portlet name in the default language too may have changed. You need to verify this.

   ---

7. **Select 'My Account' from the welcome dock.**

8. **Click 'Display Settings' under 'Miscellaneous'.**

9. **Select a language from the 'Language' list which is the same as the language selected previously.**

   In this example, select French.

10. **Click Save and navigate back to the community page.**

    You can see the changed name of the portlet in the localized version. In this example, the name of the Admin portlet localized to French is changed from `Administration du portail` to `Administration`.

CHAPTER 4

Administering Roles and Permissions

Roles and permissions can be administered by using portlets such as the *Enterprise Admin Portlet* and *Communities Portlet*. Alternatively, you can use the *Control Panel* to administer roles and permissions. The admin user needs to administer roles and permissions for the resources he need to manage to build a site.

- Portlets for Administering Roles and Permissions
- "WebSynergy Portal Resources" on page 69
- User Groups
- Communities
- Organizations
- Users and User Roles
- Plugins

Portlets for Administering Roles and Permissions

The admin user can use the Control Panel to set roles and permissions for users, user groups, communities, and organizations. Also, the Enterprise Admin portlet and Communities are the most helpful in administering roles and permissions.

- "Using the Control Panel to Administer Roles and Permissions" on page 65
- "Using Administrative Portlets of WebSynergy" on page 68

Using the Control Panel to Administer Roles and Permissions

The admin user can use the Control Panel to administer roles and permissions.

▼ To use the Control Panel to Administer Roles and Permissions

1. **Login to Project WebSynergy as Admin user.**

2. **Choose Control Panel from the welcome menu.**

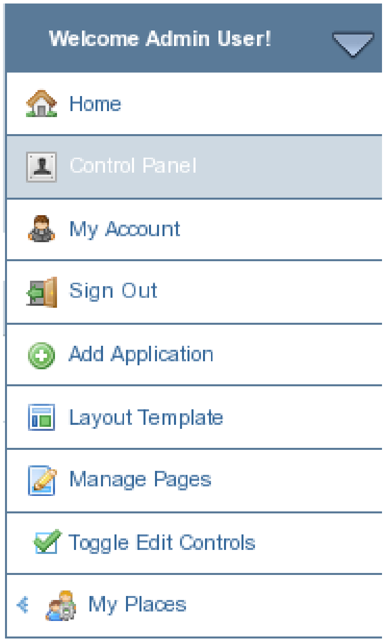
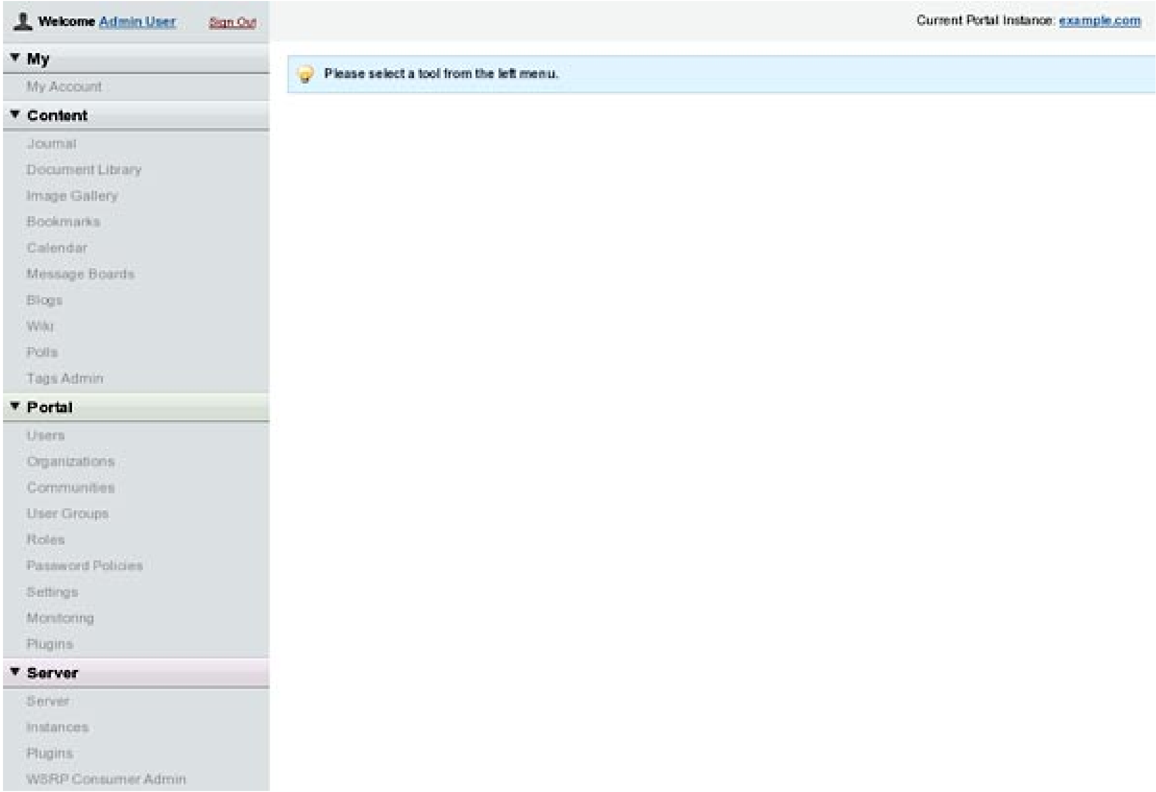


FIGURE 4–1 Choosing Control Panel from the welcome menu

The Control Panel page appears.



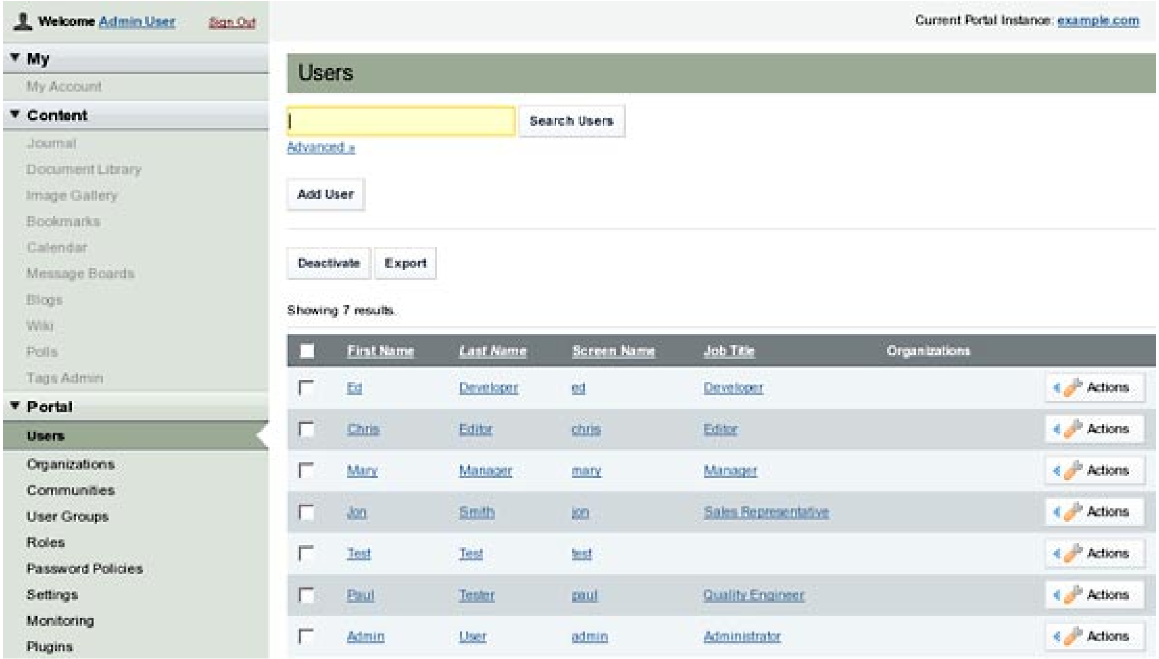
Current Portal Instance: example.com

Please select a tool from the left menu.

FIGURE 4–2 The Control Panel

3 To view and make changes to an item, choose an item from the left menu.

For example, click *Users* under *Portal*. From this page, you can set roles and permissions for users. Similarly, you can set roles and permissions for *Organizations*, *Communities*, and *User Groups*.



Showing 7 results.

☐	First Name	Last Name	Screen Name	Job Title	Organizations	
☐	Ed	Developer	ed	Developer		Actions
☐	Chris	Editor	chris	Editor		Actions
☐	Mary	Manager	mary	Manager		Actions
☐	Jon	Smith	jon	Sales Representative		Actions
☐	Test	Test	test			Actions
☐	Paul	Tester	paul	Quality Engineer		Actions
☐	Admin	User	admin	Administrator		Actions

FIGURE 4–3 Using the Control Panel

Using Administrative Portlets of WebSynergy

The *Enterprise Admin* portlet is used for most administrative tasks. This portlet has an interface for the creation and maintenance of the following:

- Users
- Organizations
- User Groups
- Roles

Additionally, it allows you to configure many server settings, including:

- Information about the site
- Authentication options, including Single Sign-On and LDAP integration
- Default User Associations
- Reserved Screen Names
- Mail Host Names
- Email Notifications

You need to use the *Enterprise Admin* portlet to create your portal structure, implement security, and administer users. Only the users with the Administrator role has permission to add the Enterprise Admin portlet to a page.

WebSynergy Portal Resources

The following is a brief overview of concepts which WebSynergy uses to organize a portal and to add and maintain user resources.

1. Portals are accessed by *Users*.
2. *Users* can be collected into *User Groups*.
3. *Users* can belong to *Organizations*.
4. *Organizations* can be grouped into hierarchies.
5. *Users*, *Groups*, and *Organizations* can belong to *Communities* that have a common interest.

The simplest way to think about this is that you have users and various ways those users can be grouped together. Some of these groupings follow an administratively organized hierarchy, and other groupings may be done by the users themselves (such as different users from multiple organizations starting a community called 'Dog Lovers' that has a common interest in dogs). And other groupings may be done administratively via User Groups or Roles for other functions that may cut across the portal (such as a Message Board Administrators group made up of users from multiple communities and organizations, allowing those users to administer any message board in the portal).

This way of organizing portal concepts may be illustrated in the following manner:

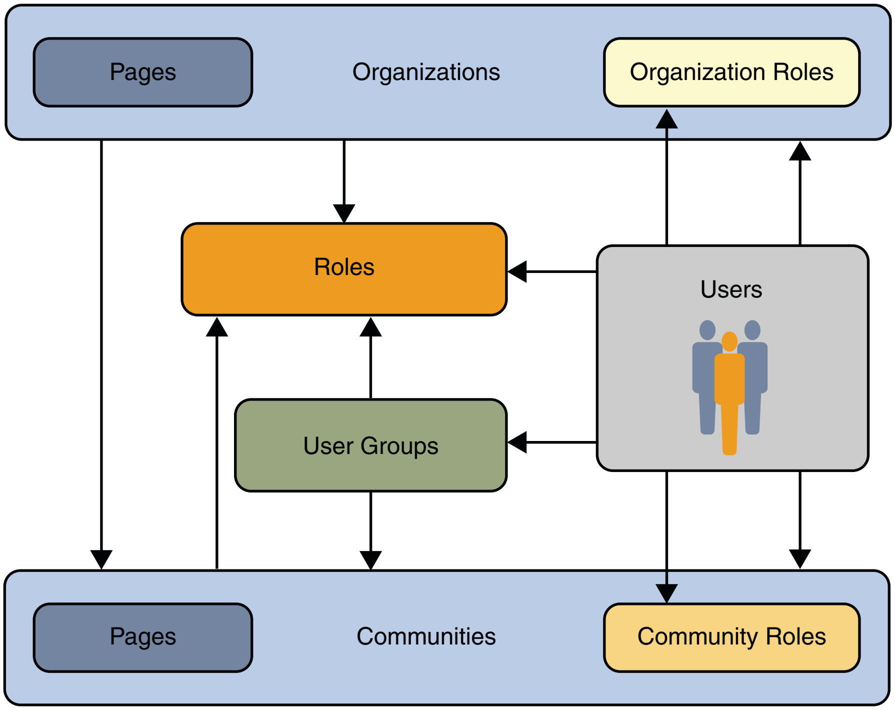


FIGURE 4–4 WebSynergy Portal Resources

In the illustration above, each arrow may be read using the words “can be a member of”. So this means that Organizations can be members of Communities, Communities can be members of Roles, Users can be members of anything, and so on. Though this seems very complex, it provides a powerful mechanism for portal administrators to configure portal resources and security in a consistent and robust manner.

User Groups

User Groups are arbitrary groupings of users. These groups are created by portal administrators to group users who do not have an obvious organizational or community-based attribute or aspect that brings them together. Groups can have permissions, much like roles. You would therefore use a User Group to grant permissions to any arbitrary list of users.

For example, a User Group called “People Who Have Access to My Stuff” could be created, and permission to a particular Document Library folder could be granted to that User Group. This list of users could be members of separate Organizations, Communities, or Roles, who happen to also have access to this 'Document Library' folder which is on some personal, community, or organization page that is accessible to them in the portal.

- Defining User Groups

- Assigning Users to User Groups

Defining User Groups

To be done.

▼ To Create a User Group

1. **Login to WebSynergy as Admin user.**

2. **Click 'Add Applications' from the welcome menu.**

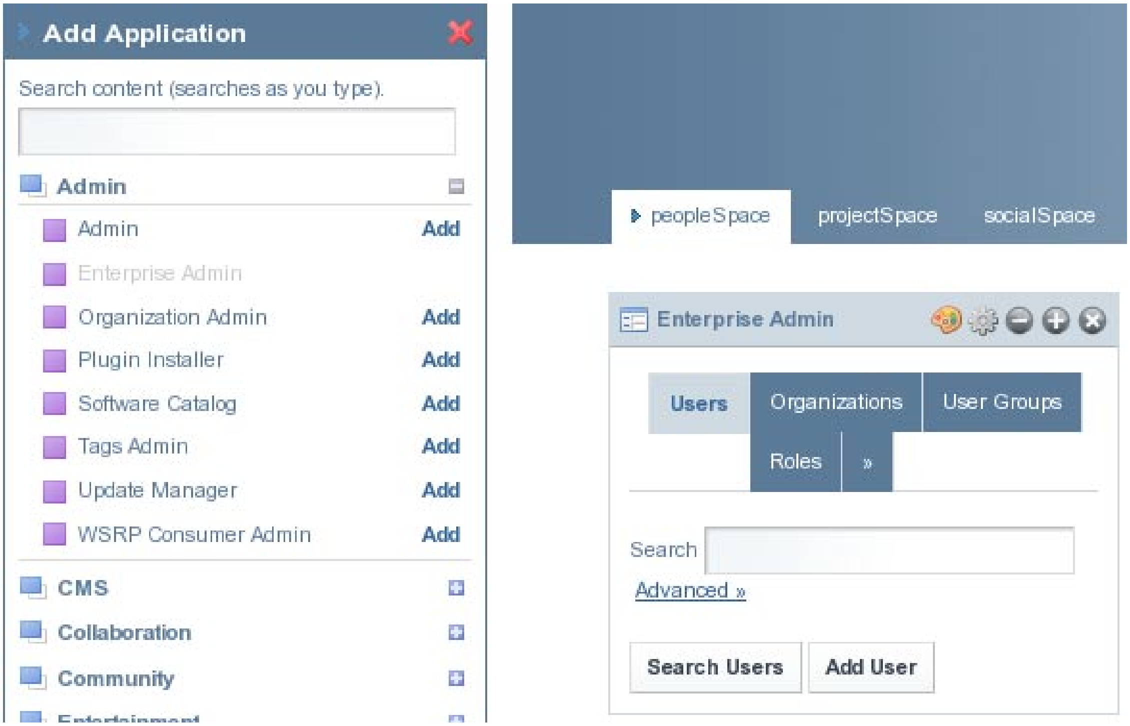


FIGURE 4–5 Adding Enterprise Admin portlet to the portlet page

3. **Click Add' against the 'Enterprise Admin' portlet.**

4. **Click on 'User Groups'.**

5. **Click 'Add'.**

6 **Specify a name and a description for the group and click Save.**

Assigning Users to User Groups

To be done.

▼ To Assign Users to User Groups

1 **Login to WebSynergy as Admin user.**

2 **Click 'Add Applications' from the welcome menu.**

3 **Click 'Add' against the 'Enterprise Admin' portlet.**

4 **Click on 'User Groups'.**

5 **Click 'Actions' button corresponding to a user group and select 'Assign Members' from the menu.**

This page allows making assignments to the user group.

6 **Choose the 'Available' tab to view the list of all available members.**

7 **Select the users who you are needed in the user group and click 'Update Associations'.**

8 **Choose the 'Current' tab to verify updated associations to the user group.**

Communities

Communities are collections of Users who have a common interest. WebSynergy's default pages are in the 'Guest' community, because everyone-whether they are anonymous or members of the portal has common interest in the default public pages of your site.

There are three types of Communities:

- Open
- Restricted
- Private

An Open Community (the default) allows portal users to join and leave the Community whenever they want to, provided they have access to a Communities portlet from which to do this. A Restricted Community requires that users be added to the Community by a community administrator. Users may use the 'Communities' portlet to request membership. A Private community do not allow users to join and to leave the community.

- Defining Communities
- Assigning Users to Communities

Defining Communities

To be done.

▼ To add a Community

1. **Login to WebSynergy as Admin user.**
2. **Click 'Add Applications' from the welcome menu.**
3. **Click 'Add' against the 'Communities' portlet.**
4. **Click 'Add Community'.**
5. **Specify the Name, Description, Type, and Active attributes for the community. Open is the default type.**
6. **Click Save to save the community.**

Assigning Users to Communities

To be done.

▼ To Assign Users to a Community

1. **Login to WebSynergy as Admin user.**
2. **Click 'Add Applications' from the welcome menu.**
3. **Click 'Add' against the 'Communities' portlet.**
4. **Click 'Available Communities' to list all available communities.**
5. **Click the 'Actions' button for the community, and select 'Assign Members' from the menu.**

   All the current *Users*, *Organizations*, and *User Groups* are listed in their respective tabs.
6. **Click 'Available' tab under 'Users' tab to list all available users.**
7. **Select the checkbox against each of the users who you want to assign as members.**

8 Click 'Update Associations' button to assign selected users as members.

9 Click 'Current' tab to view the current members which includes the new members assigned to the community.

Organizations

Organizations are hierarchical collections of Users. They are one of the two types of portal resources that can have pages. There is also a special type of Organization called 'Location', which can define where users are specifically located.

There are two kinds of Organizations:

- Regular
- Location

The organization of type Regular can have suborganizations, but an organization of type Location is not allowed to have suborganizations.

Let us consider the example of creating an organization called *Admin Organization* and a Suborganization called *Admin Location*.

- Defining Organizations
- Creating Users Under Organizations

Defining Organizations

This section outlines the procedure for Create an Organizationand Creating a Suborganization.

▼ To Create an Organization

1 Login to WebSynergy as Admin user.

2 Click 'Add Applications' from the welcome menu.

3 Click 'Add' against the 'Enterprise Admin' portlet.

4 Click 'Organizations'.

All the existing organizations are listed.

5 Click 'Add' to add a new organization.

6 Specify the 'Name' and 'Type' for the organization. For an organization of type 'Location', also specify the 'Country' and 'Region'.

7 **Click Save.**

You can specify the other attributes such as, address, phone number, email ID, and website.

8 **Click 'Return to Full Page' to navigate back.**

9 **Click the 'Organizations' tab.**

You will find the just created organization in the list of all organizations.

▼ To Create a Suborganization

1 **Login to WebSynergy as Admin user.**

2 **Click 'Add Applications' from the welcome menu.**

3 **Click 'Add' against the 'Enterprise Admin' portlet.**

4 **Click 'Organizations'.**

All the existing organizations are listed.

5 **Click 'Add' to add a new organization.**

6 **Specify the 'Name' and 'Type' for the organization. For an organization of type 'Location', also specify the 'Country' and 'Region'.**

7 **Click 'Select' to select a parent organization.**

The resultant window lists all the Organizations of type Regular.

Note – The system restricts creating a Location Organization without assigning a parent organization to it.

8 **Select an organization to make it the parent organization.**

The organization being created becomes the suborganization of the parent organization.

9 **Click Save.**

Note – An alternative method for selecting a parent organization for an organization is by selecting *Add Regular Organization* or *Add Location* from the menu appearing on clicking the *Actions* button corresponding to an organization. If you want to select a *Regular Organization* as the parent organization, select *Add Regular Organization* from the menu. If you want to select an organization of type *Location* as the parent organization, select *Add Location* from the menu.

Creating Users Under an Organizations

To be done.

▼ To Create Users under an Organization

1. **Login to WebSynergy as Admin user.**

2. **Click 'Add Applications' from the welcome menu.**

3. **Click 'Add' against the 'Enterprise Admin' portlet.**

4. **Click 'Organizations'.**

   All the existing organizations are listed.

5. **Click 'Actions' button corresponding to an organization.**

6. **Select 'Add User' from the menu.**

   Specify the name and other details for the user. By default, the user is assigned to the same organization. To change the organization for the user, or to assign the user to more organizations, click the *Organizations* link under *User Information*.

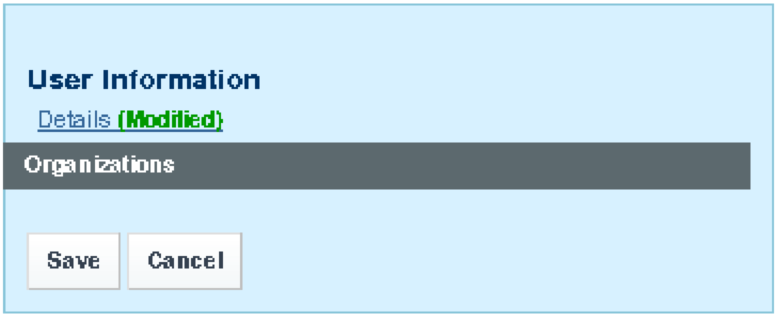


FIGURE 4–6 Selecting an organization

- **Click 'Select' to select another organization for the user.**
- **Click 'Remove' corresponding to an organization to remove user from the organization.**

7 **Click Save.**

The user is created.

8 **Update the User Information, Identification, and other details of the organization.**

9 **Click Save.**

Users and User Roles

There are three kinds of User Roles:

- Regular
- Organization
- Community

- What Different User Roles Mean?
- Defining User Roles

What Different User Roles Mean?

A user generally has some *Regular*, *Organization*, and *Community* roles. You can view the roles and other attributes associated with a user by selecting a user from the *User* tab in the *Enterprise Admin* portlet.

Regular Roles

Guest, *User*, *Power User*, *Owner*, and *Administrator* are the different *Regular Roles*. All the user with login access to WebSynergy are assigned with the 'User' role. It defines the difference between a Guest and a person who has a user ID in the portal. By default, all users are also assigned the Power User role. This role by default gives users their own personal pages (both public and private) where they can place portlets.

To make yourself an Administrator, click the *Select* link. A window listing all the roles in the system pops us. Click the *Administrator* link from the pop-up window. You are now an administrator of the portal. Log out of the portal and then log back in with your own user ID. You can now create a private page for the administration portlets and set them up in your own space.

Organization Roles

You can assign organization roles to users who are members of a organization. Organization Administrator, Organization Member, and Organization Owner are the roles which can be assigned to a user.

Community Roles

You can assign community roles to users who are members of a community. Community Administrator, Community Member, Community Owner, Content Designer, Content Editor, and Content Publisher are the roles which can be assigned to a user.

Defining User Roles

All users are listed on clicking the *Users* tab in the *Enterprise Admin portlet*. You can assign roles to a user.

▼ To Define User Roles

1. **Login to WebSynergy as Admin user.**

2. **Click 'Add Applications' from the welcome menu.**

3. **Click 'Add' against the 'Enterprise Admin' portlet.**

4. **Click the "Users' tab.**

5. **Choose a user from the list.**

   You can define *Regular Roles*, *Community Roles*, and *Organization Roles* tabs allow assigning Regular, Community, and Organization roles for the user.

6 **Click 'Roles' under 'User Information' for the user.**

You can assign *Regular* roles for all users in addition to the default Regular roles. To assign *Organization* or *Community* roles, the user need to be a member of a WebSynergy organization or community. Also you can remove the roles assigned to a user.

- **Choose 'Select' under Regular Roles, Organization Roles, or Community Roles to assign more Regular, Organization, or Community roles respectively.**
- **Choose 'Remove' corresponding to a role to remove the role assigned.**

7 **Click Save.**

Plugins

To be done.

- What Are Plugins?
- Enabling Plugins
- Working With Plugin Permissions

What Are Plugins?

To be done.

Enabling Plugins

To be done.

▼ To Access Plugins

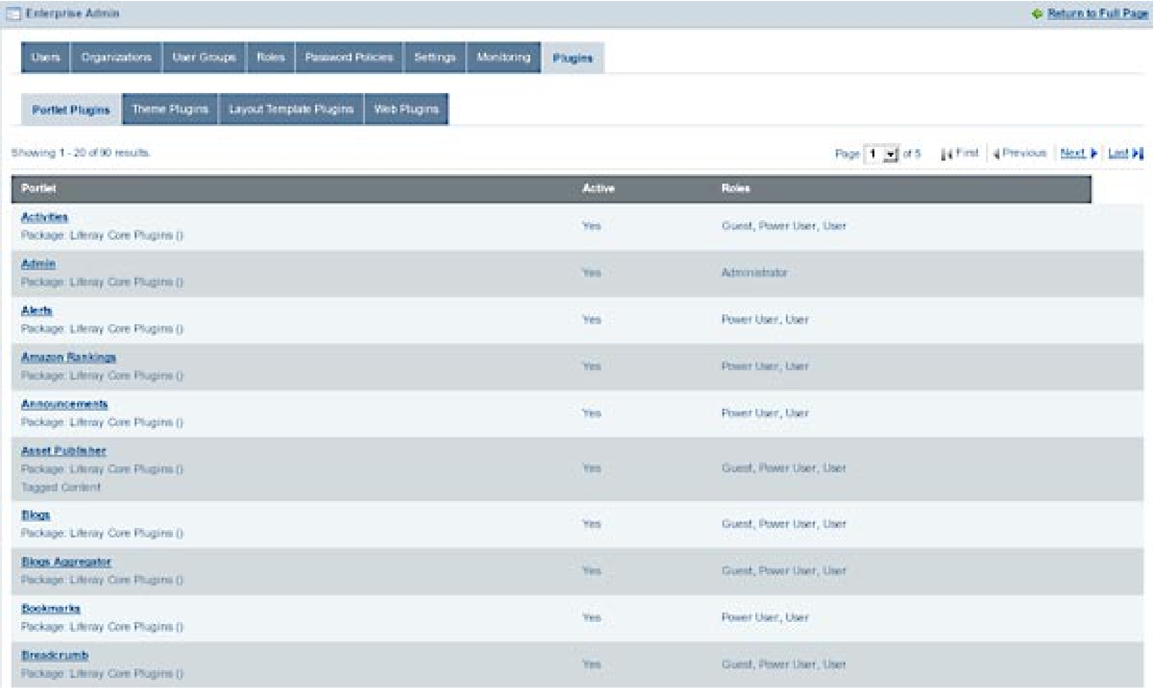
1 **Login to WebSynergy as Admin user.**

2 **Click 'Add Applications' from the welcome menu.**

3 **Click 'Add' against the 'Enterprise Admin' portlet.**

4 **Expand the Enterprise Admin portlet to view all the tabs associated with the portlet.**

5 **Choose the 'Plugins' tab.**



Showing 1 - 20 of 90 results.

Portlet	Active	Roles
Activities Package: Liferay Core Plugins ()	Yes	Guest, Power User, User
Admin Package: Liferay Core Plugins ()	Yes	Administrator
Alerts Package: Liferay Core Plugins ()	Yes	Power User, User
Amazon Rankings Package: Liferay Core Plugins ()	Yes	Power User, User
Announcements Package: Liferay Core Plugins ()	Yes	Power User, User
Asset Publisher Package: Liferay Core Plugins () Tagged Content	Yes	Guest, Power User, User
Blogs Package: Liferay Core Plugins ()	Yes	Guest, Power User, User
Blogs Aggregator Package: Liferay Core Plugins ()	Yes	Guest, Power User, User
Bookmarks Package: Liferay Core Plugins ()	Yes	Power User, User
Breadcrumb Package: Liferay Core Plugins ()	Yes	Guest, Power User, User

FIGURE 4–7 Plugins tab of the Enterprise Admin portlet

You can see four tabs listing portlets of four different categories namely; *Portlet Plugins*, *Theme Plugins*, *Layout Template Plugins*, *Web Plugins*.

Working With Plugin Permissions

You can activate or deactivate a Portlet Plugin, and you can change the User Roles associated with a plugin. For example, the Admin portlet has the Administrator role associated with it. This means, a user registered as an administrator can access the portlet.

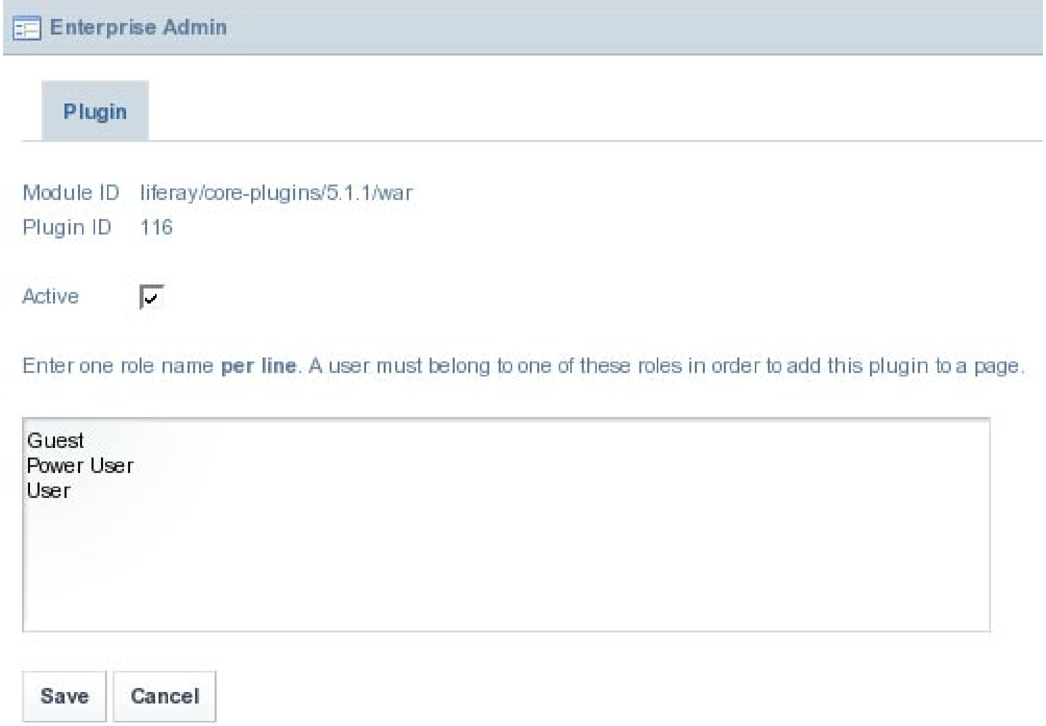
▼ To Change Plugin Permissions

1. **Login to WebSynergy as Admin user.**
2. **Click 'Add Applications' from the welcome menu.**
3. **Click 'Add' against the 'Enterprise Admin' portlet.**
4. **Expand the Enterprise Admin portlet to view all the tabs associated with the portlet.**

5 **Choose the 'Plugins' tab.**

6 **Click the link for a portlet.**

For example, click on the *Activities* portlet.



Enterprise Admin

Plugin

Module ID liferay/core-plugins/5.1.1/war

Plugin ID 116

Active ☑

Enter one role name **per line**. A user must belong to one of these roles in order to add this plugin to a page.

Guest
Power User
User

FIGURE 4–8 Changing the plugin permissions

- **Change the user access permissions for the portlet by adding or removing roles.**

  For example, If the portlet has 'Power User' role, all the users who have this role can access the portlet.

- **Activate or deactivate the portlet with the 'Active' option.**

7 **Click Save.**

CHAPTER 5

Advanced WebSynergy Configuration

WebSynergy is configured by a combination of settings which are stored in the database (configured by the use of the various administration portlets) and settings which are stored in properties (text) files. These files can be modified to change the portals behavior in certain ways. There are a large number of configuration options that can be set, and so this section will have a wide-ranging set of topics. We will first go over the main configuration file, which is stored in *<path to Glassfish Dir>*`/domains/domain1/application/j2ee-modules/websynergy/WEB-INF/classes`, and is called `portal-ext.properties`.

There are also some lower level settings that you may want to further customize. They include changing certain out-of-box defaults, security configuration, and adding features to WebSynergy through plugin management.

The *portal-ext.properties* File

WebSynergy's properties files differ from the configuration files of most other products in that changing the default configuration file is discouraged. In fact, the file that contains all of the defaults is stored inside of a `.jar` file, making it more difficult to customize. Why is it set up this way? Because WebSynergy uses the concept of overriding the defaults in a separate file, rather than going in and customizing the default configuration file. You put just the settings you want to customize in your own configuration file, and then the configuration file for your portal is uncluttered and contains only the settings you need. This makes it far easier to determine whether a particular setting has been customized, and it makes the settings more portable across different instances of WebSynergy.

The default configuration file is called `portal.properties`, and it resides inside of the `portal-impl.jar` file. This file is readily accessible inside of the <*WebSynergy home*>`/WEB-INF/classes` folder in WebSynergy's installed location in your application server. What follows is a brief description of the options that can be placed in the `portal-ext.properties` file, thus overriding the defaults from the `portal.properties` file. These are presented in a logical order, not an alphabetical one, as many properties relate to other properties in the system.

You need to create the `portal-ext.properties` file by placing an option you want to override and place it in the <*WebSynergy home*>`/WEB-INF/classes` folder to override that option in `portal.properties`.

Note – If you are running on GlassFish, WebSynergy's installed location is <*path to Glassfish Dir*>`/domains/domain1/application/j2ee-modules/websynergy/`.

Properties Override

This property specifies where to get the overridden properties. By default, it is `portal-ext.properties`. Updates should not be made on the original file (`portal.properties`) but on the overridden version of this file. Furthermore, each portal instance can have its own overridden property file following the convention `portal-companyid.properties`.

For example, one read order may be: `portal.properties`, then `portal-ext.properties`, and then `portal-test.properties`.

Examples:

```
include-and-override=portal-ext.properties
```

```
include-and-override=portal-${easyconf:companyId}.properties
```

```
include-and-override=portal-test.properties
```

You can add additional property files that overwrite the default values by using the `external-properties` system property.

A common use case is to keep legacy property values when upgrading to newer versions of WebSynergy. For example:

```
java ... -Dexternal-properties=portal-legacy-4.4.properties
include-and-override=${external-properties}
```

Portal Context

This specifies the path of the portal servlet context. This is needed because `javax.servlet.ServletContext` does not have access to the context path until Java EE 5.

Set this property if you deploy the portal to another path besides root.

Examples:

```
portal.ctx=/
portal.ctx=/portal
```

Resource Repositories Root

Specifies the default root path for various repository and resource paths. Under this path several directories will be created for the hot deploy feature, JCR, etc.

Examples:

```
resource.repositories.root=${user.home}/liferay
resource.repositories.root=/home/liferay
```

Technology Compatibility Kit

Set the following to true to enable programmatic configuration to let the Portlet TCK obtain a URL for each test. This should never be set to true unless you are running the TCK tests.

```
tck.url=false
```

Schema

Set this to true to automatically create tables and populate with default data if the database is empty.

```
schema.run.enabled=true
```

Set this to to true to populate with the minimal amount of data. Set this to false to populate with a larger amount of sample data.

```
schema.run.minimal=true
```

Upgrade

Input a list of comma delimited class names that implement `com.liferay.portal.upgrade.UpgradeProcess`. These classes will run on startup to upgrade older data to match with the latest version.

```
upgrade.processes=\
com.liferay.portal.upgrade.UpgradeProcess_4_3_0,\
com.liferay.portal.upgrade.UpgradeProcess_4_3_1,\
com.liferay.portal.upgrade.UpgradeProcess_4_3_2,\
com.liferay.portal.upgrade.UpgradeProcess_4_3_3,\
com.liferay.portal.upgrade.UpgradeProcess_4_3_4,\
com.liferay.portal.upgrade.UpgradeProcess_4_3_5,\
com.liferay.portal.upgrade.UpgradeProcess_4_4_0,\
com.liferay.portal.upgrade.UpgradeProcess_5_0_0,\
com.liferay.portal.upgrade.UpgradeProcess_5_1_0
```

Verify

Input a list of comma delimited class names that implement `com.liferay.portal.integrity.VerifyProcess`. These classes will run on startup to verify and fix any integrity problems found in the database.

```
verify.processes=com.liferay.portal.verify.VerifyProcessSuite
```

Specify the frequency for verifying the integrity of the database.

Constants in VerifyProcess:

public static final int ALWAYS = -1;

public static final int NEVER = 0;

public static final int ONCE = 1;

```
verify.frequency=1
```

Auto Deploy

Input a list of comma delimited class names that implement `com.liferay.portal.kernel.deploy.auto.AutoDeployListener`. These classes are used to process the auto deployment of WARs.

```
auto.deploy.listeners=\
com.liferay.portal.deploy.auto.LayoutTemplateAutoDeployListener,\
com.liferay.portal.deploy.auto.PortletAutoDeployListener,\
com.liferay.portal.deploy.auto.ThemeAutoDeployListener,\
com.liferay.portal.deploy.auto.WebAutoDeployListener,\
com.liferay.portal.deploy.auto.exploded.tomcat.LayoutTemplateExplodedTomcatListener,\
com.liferay.portal.deploy.auto.exploded.tomcat.PortletExplodedTomcatListener,\
com.liferay.portal.deploy.auto.exploded.tomcat.ThemeExplodedTomcatListener
```

Set the following to true to enable auto deploy of layout templates, portlets, and themes.

```
auto.deploy.enabled=true
```

Set the directory to scan for layout templates, portlets, and themes to auto deploy.

```
auto.deploy.deploy.dir=${resource.repositories.root}/deploy
```

Set the directory where auto deployed WARs are copied to. The application server or servlet container must know to listen on that directory.

Different containers have different hot deploy paths. For example, Tomcat listens on `${catalina.base}/webapps` whereas JBoss listens on `${jboss.server.home.dir}/deploy`. Set a blank directory to automatically use the application server specific directory.

Examples:

```
auto.deploy.dest.dir=
auto.deploy.default.dest.dir=../webapps
auto.deploy.geronimo.dest.dir=${org.apache.geronimo.base.dir}/deploy
auto.deploy.geronimo-jetty.dest.dir=${org.apache.geronimo.base.dir}/deploy
auto.deploy.geronimo-tomcat.dest.dir=${org.apache.geronimo.base.dir}/deploy
auto.deploy.glassfish.dest.dir=${com.sun.aas.instanceRoot}/autodeploy
auto.deploy.glassfish-tomcat.dest.dir=${com.sun.aas.instanceRoot}/autodeploy
auto.deploy.jboss-jetty.dest.dir=${jboss.server.home.dir}/deploy
auto.deploy.jboss-tomcat.dest.dir=${jboss.server.home.dir}/deploy
auto.deploy.jetty.dest.dir=${jetty.home}/webapps
auto.deploy.jonas-jetty.dest.dir=${jonas.base}/webapps/autoload
auto.deploy.jonas-tomcat.dest.dir=${jonas.base}/webapps/autoload
auto.deploy.tomcat.dest.dir=${catalina.base}/webapps
auto.deploy.weblogic.dest.dir=${env.DOMAIN_HOME}/autodeploy
```

Set the interval in milliseconds on how often to scan the directory for changes.

```
auto.deploy.interval=10000
```

Set the number of attempts to deploy a file before blacklisting it.

```
auto.deploy.blacklist.threshold=10
```

Set the following to true if deployed WARs are unpacked. Set this to false if your application server has concurrency issues with deploying large WARs.

```
auto.deploy.unpack.war=true
```

Set the following to true if you want the deployer to rename `portlet.xml` to `portlet-custom.xml`. This is only needed when deploying the portal on WebSphere 6.1.x with a version before 6.1.0.7 because WebSphere's portlet container will try to process a portlet at the same time that WebSynergy is trying to process a portlet.

Note that according to IBM, on versions after 6.1.0.9, you need to add a context parameter to the `web.xml` descriptor in your portlet application called `com.ibm.websphere.portletcontainer.PortletDeploymentEnabled` and set it to false. This parameter causes WebSphere's built-in portlet container to ignore your portlet application when it is deployed, enabling WebSynergy to pick it up.

```
auto.deploy.custom.portlet.xml=false
```

Set this to 1 if you are using JBoss' `PrefixDeploymentSorter`. This will append a 1 in front of your WAR name. For example, if you are deploying a portlet called `testportlet.war`, it will deploy it to `1test-portlet.war`. JBoss now knows to load this portlet after the other WARs have loaded; however, it will remove the 1 from the context path.

Modify `/server/default/conf/jboss-service.xml`.

See `org.jboss.deployment.scanner.PrefixDeploymentSorter`.

```
auto.deploy.jboss.prefix=1
```

Set the path to Tomcat's configuration directory. This property is used to auto deploy exploded WARs. Tomcat context XML files found in the auto deploy directory will be copied to Tomcat's configuration directory. The context XML file must have a docBase attribute that points to a valid WAR directory.

```
auto.deploy.tomcat.conf.dir=../conf/Catalina/localhost
```

Set the path to Tomcat's global class loader. This property is only used by Tomcat in a standalone environment.

```
auto.deploy.tomcat.lib.dir=../common/lib/ext
```

Set the URLs of Libraries that might be needed to download during the auto deploy process.

```
library.download.url.quercus.jar=http://lportal.svn.sourceforge.net/viewvc/*
checkout*/lportal/portal/trunk/lib/development/quercus.jar
library.download.url.resin-util.jar=http://lportal.
svn.sourceforge.net/viewvc/*checkout*/lportal/portal/trunk/lib/development/
resin-util.jar
library.download.url.script-10.jar=http://lportal.svn.sourceforge.net/viewvc
/*checkout*/lportal/portal/trunk/lib/development/script-10.jar
```

Hot Deploy

Input a list of comma delimited class names that implement `com.liferay.portal.kernel.deploy.hot.HotDeployListener`. These classes are used to process the deployment and undeployment of WARs at runtime.

Note – `PluginPackageHotDeployListener` must always be first.

```
hot.deploy.listeners=\
com.liferay.portal.deploy.hot.PluginPackageHotDeployListener,\
com.liferay.portal.deploy.hot.HookHotDeployListener,\
com.liferay.portal.deploy.hot.LayoutTemplateHotDeployListener,\
com.liferay.portal.deploy.hot.PortletHotDeployListener,\
com.liferay.portal.deploy.hot.ThemeHotDeployListener,\
com.liferay.portal.deploy.hot.ThemeLoaderHotDeployListener
```

Hot Undeploy

Set the following to true to enable undeploying plugins.

```
hot.undeploy.enabled=true
```

Set the undeploy interval in milliseconds on how long to wait for the undeploy process to finish.

```
hot.undeploy.interval=0
```

Set the following to true to undeploy a plugin before deploying a new version. This property will only be used if the property `hot.undeploy.enabled` is set to true.

```
hot.undeploy.on.redeploy=false
```

Plugin

Input a list of comma delimited supported plugin types.

```
plugin.types=portlet,theme,layout-template,web
```

Input a list of Liferay plugin repositories separated by \n characters.

```
plugin.repositories.trusted=http://plugins.liferay.com/official
plugin.repositories.untrusted=http://plugins.liferay.com/community
```

Set this property to false to avoid receiving on screen notifications when there is a new version of an installed plugin.

```
plugin.notifications.enabled=true
```

Input a list of plugin packages ids separated by \n characters. Administrators won't be notified when a new version of these plugins are available. The ids are of the form groupId/artifactId. You can also end the id with an asterisk to match any id that start with the previous character.

```
plugin.notifications.packages.ignored=liferay/sample-jsp-portlet
```

Portlet

Set this property for the portlet container implementation to use. The default implementation is the internal implementation and provides for the best backwards compatibility. The Sun implementation provides more features and will be the recommended implementation in the future.

```
portlet.container.impl=internal
#portlet.container.impl=sun
```

Set this property to define the default virtual path for all hot deployed portlets. See `liferay-portlet-app_4_3_0.dtd` and the virtual-path element for more information.

```
portlet.virtual.path=
```

Theme

Set this property to true to load the theme's merged CSS files for faster loading for production.

Set this property to false for easier debugging for development. You can also disable fast loading by setting the URL parameter `css_fast_load` to 0.

```
theme.css.fast.load=true
```

Set the theme's shorcut icon.

```
theme.shortcut.icon=liferay.ico
```

Set this property to set the default virtual path for all hot deployed themes. See `liferay-look-and-feel_4_3_0.dtd` and the virtual-path element for more information.

```
theme.virtual.path=
```

Set this with an absolute path to specify where imported theme files from a LAR will be stored. This path will override the file-storage path specified in `liferay-themeloader. xml`.

```
theme.loader.storage.path=
```

Themes can be imported via LAR files. Set this to true if imported themes should use a new theme id on every import. This will ensure that a copy of the old theme is preserved in the theme loader storage path. However, this also means that a lot of themes that are no longer used remain in the file system. It is recommended that you set this to false.

```
theme.loader.new.theme.id.on.import=false
```

Set this to true to decorate portlets by default.

```
theme.portlet.decorate.default=true
```

Resource Actions

Input a list of comma delimited resource action configurations that will be read from the class path.

```
resource.actions.configs=resource-actions/default.xml
```

Model Hints

Input a list of comma delimited model hints configurations.

```
model.hints.configs=\
META-INF/portal-model-hints.xml,\
META-INF/workflow-model-hints.xml,\
META-INF/ext-model-hints.xml,\
META-INF/portlet-model-hints.xml
```

String

Input a list of comma delimited Spring configurations. These will be loaded after the bean definitions specified in the `contextConfigLocation` parameter in `web.xml`.

```
spring.configs=\
META-INF/data-source-spring.xml,\
META-INF/misc-spring.xml,\
META-INF/counter-spring.xml,\
META-INF/documentlibrary-spring.xml,\
META-INF/lock-spring.xml,\
META-INF/mail-spring.xml,\
META-INF/portal-spring.xml,\
META-INF/portal-spring-jcr.xml,\
META-INF/ext-spring.xml
```

Set the bean name for the Liferay data source.

```
spring.hibernate.data.source=liferayDataSource
```

Set the bean name for the Liferay session factory.

```
spring.hibernate.session.factory=&liferaySessionFactory
```

Hibernate

Many of the following properties should only be customized if you have advanced knowledge of Hibernate. They map to various Hibernate configuration options which themselves have detailed documentation. Please see http://www.hibernate.org for more information.

Input a list of comma delimited Hibernate configurations.

```
hibernate.configs=\
META-INF/counter-hbm.xml,\
META-INF/mail-hbm.xml,\
META-INF/portal-hbm.xml,\
META-INF/ext-hbm.xml
```

Use the Liferay SQL dialect because it will automatically detect the proper SQL dialect based on your connection URL.

```
hibernate.dialect=com.liferay.portal.dao.orm.hibernate.DynamicDialect
```

Set the Hibernate connection release mode. You should not modify this unless you know what you're doing. The default setting works best for Spring managed transactions. See the method `buildSessionFactory` in class `org.springframework.orm.hibernate3.LocalSessionFactoryBean` and search for the phrase "on_close" to understand how this works.

```
hibernate.connection.release_mode=on_close
```

Set the Hibernate cache provider. Ehcache is recommended in a clustered environment. See the property `net.sf.ehcache.configurationResourceName` for detailed configuration.

Examples:

```
hibernate.cache.provider_class=com.liferay.portal.dao.orm.hibernate.EhCacheProvider
hibernate.cache.provider_class=net.sf.hibernate.cache.HashtableCacheProvider
hibernate.cache.provider_class=com.liferay.portal.dao.orm.hibernate.OSCacheProvider
```

This property is used if Hibernate is configured to use Ehcache's cache provider.

```
net.sf.ehcache.configurationResourceName=/ehcache/hibernate.xml
```

Use the following ehcache configuration in a clustered environment.

```
net.sf.ehcache.configurationResourceName=/ehcache/hibernate-clustered.xml
```

Set other Hibernate cache settings.

```
hibernate.cache.use_query_cache=true
hibernate.cache.use_second_level_cache=true
hibernate.cache.use_minimal_puts=true
hibernate.cache.use_structured_entries=false
```

Use these properties to disable Hibernate caching. This may be a performance hit; you may only want to use these properties for diagnostic purposes.

```
hibernate.cache.provider_class=org.hibernate.cache.NoCacheProvider
hibernate.cache.use_query_cache=false
hibernate.cache.use_second_level_cache=false
```

Set the JDBC batch size to improve performance. If you're using Oracle 9i, however, you must set the batch size to 0 as a workaround for a hanging bug in the Oracle driver. See http://support.liferay.com/browse/LEP-1234 for more information.

Examples:

```
hibernate.jdbc.batch_size=20
hibernate.jdbc.batch_size=0
```

Set other miscellaneous Hibernate properties.

```
hibernate.jdbc.use_scrollable_resultset=true
hibernate.bytecode.use_reflection_optimizer=true
hibernate.show_sql=false
```

Use the classic query factory until WebLogic and Hibernate 3 can get along. See http://www . hibernate .org/250 . html #A23 for more information.

```
hibernate.query.factory_class=org.hibernate.hql.classic.ClassicQueryTranslatorFactory
```

Ehcache

Set the classpath to the location of the Ehcache config file for internal caches. Edit the file specified in the property `ehcache.multi-vm.config.location` to enable clustered cache.

```
ehcache.single.vm.config.location=/ehcache/liferay-single-vm.xml
ehcache.multi.vm.config.location=/ehcache/liferay-multi-vm.xml
```

Use the following in a clustered environment.

```
ehcache.multi.vm.config.location=/ehcache/liferay-multi-vm-clustered.xml
```

Commons Pool

Commons Pool is used to pool and recycle objects that are used very often. This can help lower memory usage. There is some debate over the synchronization issues related to Commons Pool. Set this to false to disable object pooling.

```
commons.pool.enabled=false
```

JavaScript

Set a list of JavaScript files that will be loaded programmatically in `/html/common/themes/top_js.jsp`.

There are two lists of files specified in the properties `javascript.barebone.files` and `javascript.everything.files`.

As the name suggests, the barebone list is a trimmed down version of the everything list whereas the everything list is a list of all loaded JavaScript files.

The two lists of files exist for performance reasons because unauthenticated users usually do not utilize all the JavaScript that is available. See the property `javascript.barebone.enabled` for more information on the logic of when the barebone list is used and when the everything list is used and how to customize that logic.

The list of files are also merged and packed for further performance improvements. See the property `javascript.fast.load` for more details.

Specify the list of barebone files.

The ordering of the JavaScript files is important. Specifically, all JQuery scripts should go first.

The Liferay scripts are grouped in such a way that the first grouping denotes utility scripts that are used by the second and third groups. The second grouping denotes utility classes that rely on the first group, but does not rely on the second or third group. The third grouping denotes modules that rely on the first and second group.

```
javascript.barebone.files=\
\
#
# JQuery scripts
#
\
jquery/jquery.js,\
jquery/cookie.js,\
jquery/hover_intent.js,\
jquery/j2browse.js,\
jquery/livequery.js,\
jquery/ui.core.js,\
jquery/ui.datepicker.js,\
jquery/ui.dialog.js,\
\
#
# Miscellaneous scripts
#
\
misc/class.js,\
misc/swfobject.js,\
\
#
# Liferay base utility scripts
#
\
liferay/liferay.js,\
liferay/browser.js,\
liferay/util.js,\
\
#
# Liferay utility scripts
#
\
liferay/events.js,\
liferay/popup.js,\
liferay/portal.js,\
liferay/portlet.js,\
\
#
# Liferay modules
#
\
liferay/dock.js,\
liferay/menu.js
#
# Specify the list of everything files.
#
javascript.everything.files=\
\
#
# JQuery scripts
#
\
jquery/jquery.js,\
jquery/cookie.js,\
jquery/form.js,\
jquery/hover_intent.js,\
jquery/j2browse.js,\
jquery/jeditable.js,\
jquery/json.js,\
jquery/livequery.js,\
jquery/media.js,\
jquery/ui.core.js,\
jquery/ui.accordion.js,\
jquery/ui.dialog.js,\
jquery/ui.draggable.js,\
jquery/ui.droppable.js,\
jquery/ui.resizable.js,\
jquery/ui.selectable.js,\
jquery/ui.slider.js,\
jquery/ui.sortable.js,\
jquery/ui.tabs.js,\
jquery/effects.core.js,\
jquery/effects.blind.js,\
jquery/effects.bounce.js,\
jquery/effects.clip.js,\
jquery/effects.drop.js,\
jquery/effects.explode.js,\
jquery/effects.fold.js,\
jquery/effects.highlight.js,\
jquery/effects.pulsate.js,\
jquery/effects.scale.js,\
jquery/effects.shake.js,\
jquery/effects.slide.js,\
jquery/effects.transfer.js,\
jquery/ui.color_picker.js,\
jquery/ui.autocomplete.js,\
\
#
# Miscellaneous scripts
#
\
misc/class.js,\
misc/swfobject.js,\
\
Advanced Liferay Configuration
#
# Liferay base utility scripts
#
\
liferay/liferay.js,\
liferay/browser.js,\
liferay/util.js,\
liferay/language.js,\
liferay/layout.js,\
\
#
# Liferay utility scripts
#
\
liferay/events.js,\
liferay/popup.js,\
liferay/portal.js,\
liferay/portlet.js,\
\
#
# Liferay modules
#
\
liferay/auto_fields.js,\
liferay/color_picker.js,\
liferay/dock.js,\
liferay/dynamic_select.js,\
liferay/layout_configuration.js,\
liferay/layout_exporter.js,\
liferay/menu.js,\
liferay/notice.js,\
liferay/navigation.js,\
liferay/session.js,\
liferay/tags_selector.js,\
liferay/upload.js
```

Set this property to false to always load JavaScript files listed in the property `javascript.everything.files`. Set this to true to sometimes load `javascript.barebone.files` and sometimes load `javascript.everything.files`.

The default logic is coded in `com.liferay.portal.events.ServicePreAction` in such a way that unauthenticated users get the barebone list of JavaScript files whereas authenticated users get the everything list of JavaScript files.

```
javascript.barebone.enabled=true
```

Set this property to true to load the packed version of files listed in the properties `javascript.barebone.files` or `javascript.everything.files`.

Set this property to false for easier debugging for development. You can also disable fast loading by setting the URL parameter `js_fast_load` to 0.

```
javascript.fast.load=true
```

Set the following to true to enable the display of JavaScript logging.

```
javascript.log.enabled=false
```

SQL Data

Set the default SQL IDs for common objects.

```
sql.data.com.liferay.portal.model.Country.country.id=19
sql.data.com.liferay.portal.model.Region.region.id=5
sql.data.com.liferay.portal.model.ListType.account.address=10000
sql.data.com.liferay.portal.model.ListType.account.email.address=10004
sql.data.com.liferay.portal.model.ListType.contact.email.address=11003
sql.data.com.liferay.portal.model.ListType.organization.status=12017
```

Company

This sets the default web id. Omni admin users must belong to the company with this web id.

```
company.default.web.id=liferay.com
```

The portal can authenticate users based on their email address, screen name, or user id.

```
company.security.auth.type=emailAddress
company.security.auth.type=screenName
company.security.auth.type=userId
```

Set this to true to ensure users login with https.

```
company.security.auth.requires.https=false
```

Set the following to true to allow users to select the remember me feature to automatically login to the portal.

```
company.security.auto.login=true
```

Set the following to the maximum age (in number of seconds) of the browser cookie that enables the remember me feature. A value of 31536000 signifies a lifespan of one year. A value of -1 signifies a lifespan of a browser session.

Rather than setting this to 0, set the property `company.security.auto.login` to false to disable the remember me feature.

```
company.security.auto.login.max.age=31536000
```

Set the following to true to allow users to ask the portal to send them their password.

```
company.security.send.password=true
```

Set the following to true to allow strangers to create accounts and register themselves on the portal.

```
company.security.strangers=true
```

Set the following to true if strangers can create accounts with email addresses that match the company mail suffix. This property is not used unless `company.security.strangers` is also set to true.

```
company.security.strangers.with.mx=true
```

Set the following to true if strangers who create accounts need to be verified via email.

```
company.security.strangers.verify=false
```

Set the following to true to allow community administrators to use their own logo instead of the enterprise logo.

```
company.security.community.logo=true
```

Users

Set the following to false if users cannot be deleted.

```
users.delete=true
```

Set the following to true to always autogenerate user screen names even if the user gives a specific user screen name.

```
users.screen.name.always.autogenerate=false
```

Input a class name that extends `com.liferay.portal.security.auth.ScreenNameGenerator`. This class will be called to generate user screen names.

```
users.screen.name.generator=com.liferay.portal.security.auth.ScreenNameGenerator
```

Input a class name that extends `com.liferay.portal.security.auth.ScreenNameValidator`. This class will be called to validate user ids.

Examples:

```
users.screen.name.validator=com.liferay.portal.security.auth.ScreenNameValidator
users.screen.name.validator=com.liferay.portal.security.auth.LiberalScreenNameValidator
```

Set the maximum file size for user portraits. A value of 0 for the maximum file size can be used to indicate unlimited file size. However, the maximum file size allowed is set in property `com.liferay.portal.upload.UploadServletRequestImpl.max.size` found in `system.properties`.

```
users.image.max.size=307200
```

Groups and Roles

Input a list of comma delimited system group names that will exist in addition to the standard system groups. When the server starts, the portal checks to ensure all system groups exist. Any missing system group will be created by the portal.

```
system.groups=
```

Input a list of comma delimited system role names that will exist in addition to the standard system roles. When the server starts, the portal checks to ensure all system roles exist. Any missing system role will be created by the portal.

The standard system roles are: Administrator, Guest, Power User, and User. These roles cannot be removed or renamed.

```
system.roles=
```

Set the description of the Administrator system role.

```
system.role.Administrator.description=Administrators are super users who can
do anything.
```

Set the description of the Guest system role.

```
system.role.Guest.description=Unauthenticated users always have this role.
```

Set the description of the Power User system role.

```
system.role.Power.User.description=Power Users have their own public and
private pages.
```

Set the description of the User system role.

```
system.role.User.description=Authenticated users should be assigned this
role.
```

Input a list of comma delimited system community role names that will exist in addition to the standard system community roles. When the server starts, the portal checks to ensure all system community roles exist. Any missing system community role will be created by the portal.

The standard system community roles are: Community Administrator, Community Member, and Community Owner. These roles cannot be removed or renamed.

```
system.community.roles=
```

Set the description of the Community Administrator system community role.

```
system.community.role.Community.Administrator.description=Community Administrators
are super users of their community but cannot make other users into
Community Administrators.
```

Set the description of the Community Member system community role.

```
system.community.role.Community.Member.description=All users who belong to a
community have this role within that community.
```

Set the description of the Community Owner system community role.

```
system.community.role.Community.Owner.description=Community Owners are super
users of their community and can assign community roles to users.
```

Input a list of comma delimited system organization role names that will exist in addition to the standard system organization roles. When the server starts, the portal checks to ensure all system organization roles exist. Any missing system organization role will be created by the portal.

The standard system organization roles are: Organization Administrator, Organization Member, and Organization Owner. These roles cannot be removed or renamed.

```
system.organization.roles=
```

Set the description of the Organization Administrator system organization role.

```
system.organization.role.Organization.Administrator.description=Organization
Administrators are super users of their organization but cannot make other
users into Organization Administrators.
```

Set the description of the Organization Member system organization role.

```
system.organization.role.Organization.Member.description=All users who belong
to a organization have this role within that organization.
```

Set the description of the Organization Owner system organization role.

```
system.organization.role.Organization.Owner.description=Organization Owners
are super users of their organization and can assign organization roles to
users.
```

Omni admin users can administer the portal's core functionality: gc, shutdown, etc. Omni admin users must belong to the default company.

Multiple portal instances might be deployed on one application server, and not all of the administrators should have access to this core functionality. Input the ids of users who are omniadmin users.

Leave this field blank if users who belong to the right company and have the Administrator role are allowed to administer the portal's core functionality.

```
omniadmin.users=
```

Set the following to true if all users are required to agree to the terms of use.

```
terms.of.use.required=true
```

Organizations

Set the following to true if organizations must have an associated country.

```
organizations.country.required=true
```

Languages and Time Zones

Specify the available locales. Messages corresponding to a specific language are specified in properties files with file names matching that of `content/Language_*.properties`. These values can also be overridden in properties files with file names matching that of `content/Language-ext_*.properties`. Use a comma to separate each entry.

All locales must use UTF-8 encoding.

See the following links to specify language and country codes:

http://ftp.ics.uci.edu/pub/ietf/http/related/iso639.txt

http://userpage.chemie.fu-berlin.de/diverse/doc/ISO_3166.html

```
locales=ar_SA,ca_AD,ca_ES,zh_CN,zh_TW,cs_CZ,nl_NL,en_US,fi_FI,fr_FR,de_DE,el
_GR,hu_HU,it_IT,ja_JP,ko_KR,fa_IR,pt_BR,ru_RU,es_ES,sv_SE,tr_TR,vi_VN
```

Set the following to true if unauthenticated users get their preferred language from the Accept-Language header. Set the following to false if unauthenticated users get their preferred language from their company.

```
locale.default.request=false
```

Specify the available time zones. The specified ids must match those from the class `java.util.TimeZone`.

```
time.zones=\
Pacific/Midway,\
Pacific/Honolulu,\
America/Anchorage,\
America/Los_Angeles,\
America/Denver,\
America/Chicago,\
America/New_York,\
America/Puerto_Rico,\
America/St_Johns,\
America/Sao_Paulo,\
America/Noronha,\
Atlantic/Azores,\
UTC,\
Europe/Lisbon,\
Europe/Paris,\
Europe/Istanbul,\
Asia/Jerusalem,\
Asia/Baghdad,\
Asia/Tehran,\
Asia/Dubai,\
Asia/Kabul,\
Asia/Karachi,\
Asia/Calcutta,\
Asia/Katmandu,\
Asia/Dhaka,\
Asia/Rangoon,\
Asia/Saigon,\
Asia/Shanghai,\
Asia/Tokyo,\
Asia/Seoul,\
Australia/Darwin,\
Australia/Sydney,\
Pacific/Guadalcanal,\
Pacific/Auckland,\
Pacific/Enderbury,\
Pacific/Kiritimati
```

Look and Feel

Set the following to false if the system does not allow users to modify the look and feel.

```
look.and.feel.modifiable=true
```

Set the default layout template id.

```
default.layout.template.id=2_columns_ii
```

Set the default theme id for regular themes.

```
default.regular.theme.id=classic
```

Set the default color scheme id for regular themes.

```
default.regular.color.scheme.id=01
```

Set the default theme id for wap themes.

```
default.wap.theme.id=mobile
```

Set the default color scheme id for wap themes.

```
default.wap.color.scheme.id=01
```

Set the following to true if you want a change in the theme selection of the public or private group to automatically be applied to the other (i.e. if public and private group themes should always be the same).

```
theme.sync.on.group=false
```

Request

Portlets that have been configured to use private request attributes in `liferayportlet.xml` may still want to share some request attributes. This property allows you to configure which request attributes will be shared.

Set a comma delimited list of attribute names that will be shared when the attribute name starts with one of the specified attribute names. For example, if you set the value to *hello_*, *world_*, then all attribute names that start with *hello_* or *world_* will be shared.

```
request.shared.attributes=LIFERAY_SHARED_
```

Session

Specify the number of minutes before a session expires. This value is always overridden by the value set in `web.xml`.

```
session.timeout=30
```

Specify the number of minutes before a warning is sent to the user informing the user of the session expiration. Specify 0 to disable any warnings.

```
session.timeout.warning=1
```

Set the auto-extend mode to true to avoid having to ask the user whether to extend the session or not. Instead it will be automatically extended. The purpose of this mode is to keep the session open as long as the user browser is open and with a portal page loaded. It is recommended to use this setting along with a smaller session.timeout, such as 5 minutes for better performance.

```
session.timeout.auto.extend=false
```

Set this to true if the user is redirected to the default page when the session expires.

```
session.timeout.redirect.on.expire=false
```

Portlets that have been configured to use private session attributes in `liferay-portlet.xml` may still want to share some session attributes. This property allows you to configure which session attributes will be shared. Set a comma delimited list of attribute names that will be shared when the attribute name starts with one of the specified attribute names. For example, if you set the value to *hello_*, *world_*, then all attribute names that start with *hello_* or *world_* will be shared.

Note that this property is used to specify the sharing of session attributes from the portal to the portlet. This is not used to specify session sharing between portlet WARs or from the portlet to the portal.

```
session.shared.attributes=org.apache.struts.action.LOCALE,COMPANY_,USER_,LIF
ERAY_SHARED_
```

Set this to false to disable all persistent cookies. Features like automatically logging in will not work.

```
session.enable.persistent.cookies=true
```

The login process sets several cookies if persistent cookies are enabled. Set this property to set the domain of those cookies.

```
session.cookie.domain=
```

Set the following to true to invalidate the session when a user logs into the portal. This helps prevents phishing. Set this to false if you need the guest user and the authenticated user to have the same session.

```
session.enable.phishing.protection=true
```

Set the following to true to test whether users have cookie support before allowing them to sign in. This test will always fail if tck.url is set to true because that property disables session cookies.

```
session.test.cookie.support=true
```

Set the following to true to disable sessions. Doing this will use cookies to remember the user across requests. This is useful if you want to scale very large sites where the user may be sent to a different server for each request. The drawback to this approach is that you must not rely on the API for sessions provided by the servlet and portlet specs.

This feature is only available for Tomcat and requires that you set Tomcat's Manager class to `com.liferay.support.tomcat.session.SessionLessManagerBase`.

```
session.disabled=false
```

Input a list of comma delimited class names that extend `com.liferay.portal.struts.SessionAction`. These classes will run at the specified event.

```
#
# Servlet session create event
#
servlet.session.create.events=com.liferay.portal.events.SessionCreateAction
#
# Servlet session destroy event
#
servlet.session.destroy.events=com.liferay.portal.events.SessionDestroy-
Action
```

Set the following to true to track user clicks in memory for the duration of a user's session. Setting this to true allows you to view all live sessions in the Admin portlet.

```
session.tracker.memory.enabled=true
```

Set the following to true to track user clicks in the database after a user's session is invalidated. Setting this to true allows you to generate usage reports from the database. Use this cautiously because this will store a lot of usage data.

```
session.tracker.persistence.enabled=false
```

Set the following to true to convert the tracked paths to friendly URLs.

```
session.tracker.friendly.paths.enabled=false
```

Enter a list of comma delimited paths that should not be tracked.

```
session.tracker.ignore.paths=\
/portal/css_cached,\
/portal/javascript_cached,\
/portal/render_portlet,\
\
/document_library/get_file
```

JASS

Set the following to false to disable JAAS security checks. Disabling JAAS speeds up login. JAAS must be disabled if administrators are to be able to impersonate other users.

```
portal.jaas.enable=false
```

By default, `com.liferay.portal.security.jaas.PortalLoginModule` loads the correct JAAS login module based on what application server or servlet container the portal is deployed on. Set a JAAS implementation class to override this behavior.

```
portal.jaas.impl=
```

The JAAS process may pass in an encrypted password and the authentication will only succeed if there is an exact match. Set this property to false to relax that behavior so the user can input an unencrypted password.

```
portal.jaas.strict.password=false
```

Set the following to true to enable administrators to impersonate other users. JAAS must also be disabled for this feature to work.

```
portal.impersonation.enable=true
```

LDAP

Set the values used to connect to a LDAP store.

```
ldap.factory.initial=com.sun.jndi.ldap.LdapCtxFactory
ldap.base.provider.url=ldap://localhost:10389
ldap.base.dn=dc=example,dc=com
ldap.security.principal=uid=admin,ou=system
ldap.security.credentials=secret
ldap.referral=follow
```

Settings for `com.liferay.portal.security.auth.LDAPAuth` can be configured from the Admin portlet. It provides out of the box support for Apache Directory Server, Microsoft Active Directory Server, Novell eDirectory, and OpenLDAP. The default settings are for Apache Directory Server.

The `LDAPAuth` class must be specified in the property `auth.pipeline.pre` to be executed.

Encryption is implemented by `com.liferay.util.Encryptor.provider.class` in `system.properties`.

```
ldap.auth.enabled=false
ldap.auth.required=false
```

Set either bind or password-compare for the LDAP authentication method. Bind is preferred by most vendors so that you don't have to worry about encryption strategies.

```
ldap.auth.method=bind
ldap.auth.method=password-compare
```

Set the password encryption to used to compare passwords if the property `ldap.auth.method` is set to password-compare.

```
ldap.auth.password.encryption.algorithm=
ldap.auth.password.encryption.algorithm.types=MD5,SHA
```

Active Directory stores information about the user account as a series of bit fields in the `UserAccountControl` attribute.

If you want to prevent disabled accounts from logging into the portal you need to use a search filter similar to the following:

```
(&(objectclass=person)(userprincipalname=@email_address@)(!(UserAccountControl:
1.2.840.113556.1.4.803:=2)))
```

See the following links:

http://support.microsoft.com/kb/305144/

http://support.microsoft.com/?kbid=269181

```
ldap.auth.search.filter=(mail=@email_address@)
```

You can write your own class that extends `com.liferay.portal.security.ldap.AttributesTransformer` to transform the LDAP attributes before a user or group is imported to the LDAP store.

```
ldap.attrs.transformer.impl=com.liferay.portal.security.ldap.Attributes-
Transformer
```

You can write your own class that extends `com.liferay.portal.security.ldap.LDAPUser` to customize the behavior for exporting portal users to the LDAP store.

```
ldap.user.impl=com.liferay.portal.security.ldap.LDAPUser
```

When a user is exported to LDAP and the user does not exist, the user will be created with the following default object classes.

```
ldap.user.default.object.classes=top,person,inetOrgPerson,organizationalPerson
```

When importing and exporting users, the portal will use this mapping to connect LDAP user attributes and portal user variables.

```
ldap.user.mappings=screenName=cn\npassword=userPassword\nemailAddress=mail\n
firstName=givenName\nlastName=sn\njobTitle=title\ngroup=groupMembership
```

When importing groups, the portal will use this mapping to connect LDAP group attributes and portal user group variables.

```
ldap.group.mappings=groupName=cn\ndescription=description\nuser=uniqueMember
```

Settings for importing users and groups from LDAP to the portal.

```
ldap.import.enabled=false
ldap.import.on.startup=false
ldap.import.interval=10
ldap.import.user.search.filter=(objectClass=inetOrgPerson)
ldap.import.group.search.filter=(objectClass=groupOfUniqueNames)
```

Set either user or group for import method. If set to user, portal will import all users and the groups associated with those users. If set to group, the portal import all groups and the users associated those groups.

This value should be set based on how your LDAP server stores group membership information.

```
ldap.import.method=user
ldap.import.method=group
```

Settings for exporting users from the portal to LDAP. This allows a user to modify his first name, last name, etc. in the portal and have that change get pushed to the LDAP server. This will only be active if the property `ldap.auth.enabled` is also set to true. New users and groups will be created at the specified DN.

```
ldap.export.enabled=true
ldap.users.dn=ou=users,dc=example,dc=com
ldap.groups.dn=ou=groups,dc=example,dc=com
```

Set this to true to use the LDAP's password policy instead of the portal password policy.

```
ldap.password.policy.enabled=false
```

Set these values to be a portion of the error message returned by the appropriate directory server to allow the portal to recognize messages from the LDAP server. The default values will work for Fedora DS.

```
ldap.error.password.age=age
ldap.error.password.expired=expired
ldap.error.password.history=history
ldap.error.password.not.changeable=not allowed to change
ldap.error.password.syntax=syntax
ldap.error.password.trivial=trivial
ldap.error.user.lockout=retry limit
```

CAS

Set this to true to enable CAS single sign on. NTLM will work only if LDAP authentication is also enabled and the authentication is made by screen name. If set to true, then the property `auto.login.hooks` must contain a reference to the class `com.liferay.portal.security.auth.CASAutoLogin` and the filter `com.liferay.portal.servlet.filters.sso.cas.CASFilter` must be referenced in `web.xml`.

```
cas.auth.enabled=false
```

A user may be authenticated from CAS and not yet exist in the portal. Set this to true to automatically import users from LDAP if they do not exist in the portal.

```
cas.import.from.ldap=false
```

Set the default values for the required CAS URLs. Set either `cas.server.name` or `cas.service.url`. Setting `cas.server.name` allows deep linking. See LEP-4423.

```
cas.login.url=https://localhost:8443/cas-web/login
cas.logout.url=https://localhost:8443/cas-web/logout
cas.server.name=localhost:8080
cas.service.url=
#cas.service.url=http://localhost:8080/c/portal/login
cas.service.url=http://localhost:8080/c/portal/login
cas.validate.url=https://localhost:8443/cas-web/proxyValidate
```

NTLM

Set this to true to enable NTLM single sign on. NTLM will work only if LDAP authentication is also enabled and the authentication is made by screen name. If set to true, then the property "auto.login.hooks" must contain a reference to the class `com.liferay.portal.security.auth.NtlmAutoLogin` and the filter `com.liferay.portal.servlet.filters.sso.ntlm.NtlmFilter` must be referenced in `web.xml`.

```
ntlm.auth.enabled=false
ntlm.auth.domain.controller=127.0.0.1
ntlm.auth.domain=EXAMPLE
```

OpenID

Set this to true to enable OpenId authentication. If set to true, then the property `auto.login.hooks` must contain a reference to the class `com.liferay.portal.security.auth.OpenIdAutoLogin`.

```
open.id.auth.enabled=true
```

OpenSSO

These properties controls Liferay's integration with OpenSSO.

Set this to true to enable OpenSSO authentication.

```
open.sso.auth.enabled=false
```

Set the log in URL and log out URL. The first URL is the link to your OpenSSO server (which can be the same server as the one running Liferay); the second URL is the link to your Liferay Portal.

```
open.sso.login.url=http://openssohost.example.com:8080/opensso/UI/Login?
goto=http://portalhost.example.com:8080/c/portal/login
open.sso.logout.url=http://openssohost.example.com:8080/opensso/UI/Logout?
goto=http://portalhost.example.com:8080/web/guest/home
```

Set the URL to the OpenSSO service.

```
open.sso.service.url=http://openssohost.example.com:8080/opensso
```

Set the HTTP attribute name for the user's screen name.

```
open.sso.screen.name.attr=uid
```

Set the HTTP attribute name for the user's email address.

```
open.sso.email.address.attr=mail
```

Set the HTTP attribute name for the user's Common Name.

```
open.sso.first.name.attr=cn
```

Set the HTTP attribute name for the user's Surname.

```
open.sso.last.name.attr=sn
```

Authentication Pipeline

Input a list of comma delimited class names that implement `com.liferay.portal.security.auth.Authenticator`. These classes will run before or after the portal authentication begins.

The Authenticator class defines the constant values that should be used as return codes from the classes implementing the interface. If# authentication is successful, return SUCCESS; if the user exists but the passwords do not match, return FAILURE; and if the user does not exist on the system, return DNE.

Constants in Authenticator:

```
public static final int SUCCESS = 1;
public static final int FAILURE = -1;
public static final int DNE = 0;
```

In case you have several classes in the authentication pipeline, all of them have to return SUCCESS if you want the user to be able to login. If one of the authenticators returns FAILURE or DNE, the login fails.

Under certain circumstances, you might want to keep the information in the portal database in sync with an external database or an LDAP server. This can easily be achieved by implementing a class via LDAPAuth that updates the information stored in the portal user database whenever a user signs in.

Each portal instance can be configured at run time to either authenticate based on user ids or email addresses. See the Admin portlet for more information.

Available authenticators are:

```
com.liferay.portal.security.auth.LDAPAuth
```

See the LDAP properties to configure the behavior of the `LDAPAuth` class.

```
auth.pipeline.pre=com.liferay.portal.security.auth.LDAPAuth
auth.pipeline.post=
```

Set this to true to enable password checking by the internal portal authentication. If set to false, you're essentially delegating password checking is delegated to the authenticators configured in `auth.pipeline.pre` and `auth.pipeline.post` settings.

```
auth.pipeline.enable.liferay.check=true
```

Input a list of comma delimited class names that implement `com.liferay.portal.security.auth.AuthFailure`. These classes will run when a user has a failed login or when a user has reached the maximum number of failed logins.

```
auth.failure=com.liferay.portal.security.auth.LoginFailure
auth.max.failures=com.liferay.portal.security.auth.LoginMaxFailures
auth.max.failures.limit=5
```

Set the following to true if users are forwarded to the last visited path upon successful login. If set to false, users will be forwarded to their default layout page.

```
auth.forward.by.last.path=true
```

The login page reads a redirect by a parameter named redirect. If this property is set to true, then users will be redirected to the given redirect path upon successful login. If the user does not have permission to view that page, then the rule set by the property `auth.forward.by.last.path` will apply.

You can set the redirect manually from another application, by appending the redirect parameter in a url that looks like this: `/c/portal/login?redirect=%2Fgroup%2Femployees%2Fcalendar`. This url will redirect the user to the path `/group/employees/calendar` upon successful login.

```
auth.forward.by.redirect=true
```

Enter a list of comma delimited paths that can be considered part of the last visited path.

```
auth.forward.last.paths=/document_library/get_file
```

Enter a URL that will be used to login portal users whenever needed. By default, the portal's login page is used.

```
#auth.login.url=/web/guest/home
```

Enter a friendly URL of a page that will be used to login portal users whenever the user is navigating a community and authentication is needed. By default, the portal's login page or the URL set in the property `auth.login.url` is used.

```
auth.login.community.url=/login
```

Enter the name of the login portlet used in a page identified by the URL of the previous property (if one has been set). This will allow the portlet to have access to the redirect parameter and thus forward the user to the page where he was trying to access when necessary. You should leave the default value unless you have your own custom login portlet.

```
auth.login.portlet.name=58
```

Enter a list of comma delimited paths that do not require authentication.

```
auth.public.paths=\
/blogs/find_entry,\
/blogs/rss,\
/blogs/trackback,\
\
/bookmarks/open_entry,\
\
/document_library/get_file,\
\
/journal/get_article,\
/journal/get_articles,\
/journal/get_latest_article_content,\
/journal/get_structure,\
/journal/get_template,\
/journal/view_article_content,\
/journal_articles/view_article_content,\
\
/layout_management/sitemap,\
\
/message_boards/find_category,\
/message_boards/find_message,\
/message_boards/find_thread,\
/message_boards/get_message_attachment,\
/message_boards/rss,\
\
/my_places/view,\
\
/polls/view_chart,\
\
/portal/expire_session,\
/portal/extend_session,\
/portal/extend_session_confirm,\
/portal/json_service,\
/portal/logout,\
/portal/open_id_request,\
/portal/open_id_response,\
/portal/session_click,\
/portal/session_tree_js_click,\
/portal/status,\
\
/search/open_search,\
/search/open_search_description.xml,\
\
/shopping/notify,\
\
/tags/rss,\
\
/wiki/get_page_attachment,\
/wiki/rss
```

Auto Login

Input a list of comma delimited class names that implement `com.liferay.portal.security.auth.AutoLogin`. These classes will run in consecutive order for all unauthenticated users until one of them return a valid user id and password combination. If no valid combination is returned, then the request continues to process normally. If a valid combination is returned, then the portal will automatically login that user with the returned user id and password combination.

For example, `com.liferay.portal.security.auth.RememberMeAutoLogin` reads from a cookie to automatically log in a user who previously logged in while checking the *Remember Me* box.

This interface allows deployers to easily configure the portal to work with other SSO servers. See `com.liferay.portal.security.auth.CASAutoLogin` for an example of how to configure the portal with Yale's SSO server.

```
auto.login.hooks=com.liferay.portal.security.auth.CASAutoLogin,com.liferay.-
portal.security.auth.NtlmAutoLogin,com.liferay.portal.security.
auth.OpenIdAutoLogin,com.liferay.portal.security.auth.OpenSSOAutoLogin,co
m.liferay.portal.security.auth.RememberMeAutoLogin
```

Set the hosts that will be ignored for auto login.

```
auto.login.ignore.hosts=
```

Set the paths that will be ignored for auto login.

```
auto.login.ignore.paths=
```

SSO With MAC (Message Authentication Code)

To use SSO with MAC, post to an URL like:

http://localhost:8080/c/portal/login?cmd=already-registered&login=<userId|emailAddress>&password=<MAC>

Pass the MAC in the password field. Make sure the MAC gets URL encoded because it might contain characters not allowed in a URL.

SSO with MAC also requires that you set the following property in `system.properties`:

```
com.liferay.util.servlet.SessionParameters=false
```

See the following links:

`http://support.liferay.com/browse/LEP-1288` http://en.wikipedia.org/wiki/Message_authentication_code

Set the following to true to enable SSO with MAC.

```
auth.mac.allow=false
```

Set the algorithm to use for MAC encryption.

```
auth.mac.algorithm=MD5
```

Set the shared key used to generate the MAC.

```
auth.mac.shared.key=
```

Passwords

Set the following encryption algorithm to encrypt passwords. The default algorithm is SHA (SHA-1). If set to NONE, passwords are stored in the database as plain text. The SHA-512 algorithm is currently unsupported.

Examples:

```
passwords.encryption.algorithm=CRYPT
passwords.encryption.algorithm=MD2
passwords.encryption.algorithm=MD5
passwords.encryption.algorithm=NONE
passwords.encryption.algorithm=SHA
passwords.encryption.algorithm=SHA-256
passwords.encryption.algorithm=SHA-384
passwords.encryption.algorithm=SSHA
```

Digested passwords are encoded via base64 or hex encoding. The default is base64.

```
passwords.digest.encoding=base64
#passwords.digest.encoding=hex
```

Input a class name that extends `com.liferay.portal.security.pwd.BasicToolkit`. This class will be called to generate and validate passwords.

Examples:

```
passwords.toolkit=com.liferay.portal.security.pwd.PasswordPolicyToolkit
passwords.toolkit=com.liferay.portal.security.pwd.RegExpToolkit
```

If you choose to use `com.liferay.portal.security.pwd.PasswordPolicyToolkit` as your password toolkit, you can choose either static or dynamic password generation. Static is set through the property `passwords.passwordpolicytoolkit.static` and dynamic uses the class `com.liferay.util.PwdGenerator` to generate the password. If you are using LDAP password syntax checking, you will also have to use the static generator so that you can guarantee that passwords obey its rules.

Examples:

```
passwords.passwordpolicytoolkit.generator=static
passwords.passwordpolicytoolkit.generator=dynamic
passwords.passwordpolicytoolkit.static=iheartliferay
```

If you choose to use `com.liferay.portal.security.pwd.RegExpToolkit` as your password toolkit, set the regular expression pattern that will be used to generate and validate passwords.

Note that \ is replaced with \\ to work in Java.

The second pattern ensures that passwords must have at least 8 valid characters consisting of digits or letters.

Examples:

```
passwords.regexptoolkit.pattern=(?=.{4})(?:[a-zA-Z0-9]*)
passwords.regexptoolkit.pattern=(?=.{8})(?:[a-zA-Z0-9]*)
```

Set the length and key for generating passwords.

Examples:

```
passwords.regexptoolkit.charset=0123456789
passwords.regexptoolkit.charset=0123456789ABCDEFGHIJKLMNOPQRSTUVWXYZabcde-
fghijklmnopqrstuvwxyz
```

Examples:

```
passwords.regexptoolkit.length=4
passwords.regexptoolkit.length=8
```

Set the name of the default password policy.

```
passwords.default.policy.name=Default Password Policy
```

Permissions

Set the default permission checker class used by `com.liferay.portal.security.permission.PermissionCheckerFactory` to check permissions for actions on objects. This class can be overridden with a custom class that extends `com.liferay.portal.security.permission.PermissionCheckerImpl`.

```
permissions.checker=com.liferay.portal.security.permission.PermissionCheckerImpl
```

Set the algorithm used to check permissions for a user. This is useful so that you can optimize the search for different databases. See `com.liferay.portal.service.impl.PermissionLocalServiceImpl`. The default is method two.

The first algorithm uses several if statements to query the database for these five things in order. If it finds any one of them, it returns true:

- Is the user connected to one of the permissions via group or organization roles?
- Is the user associated with groups or organizations that are directly connected to one of the permissions?
- Is the user connected to one of the permissions via user roles?
- Is the user connected to one of the permissions via user group roles?
- Is the user directly connected to one of the permissions?

```
permissions.user.check.algorithm=1
```

The second algorithm (the default) does a database join and checks the permissions in one step, by calling `countByGroupsRoles`, `countByGroupsPermissions`, `countByUsersRoles`, `countByUserGroupRole`, and `countByUsersPermissions` in one method.

```
permissions.user.check.algorithm=2
```

The third algorithm checks the permissions by checking for three things. It combines the role check into one step. If it finds any of the following items, it returns true:

- Is the user associated with groups or organizations that are directly connected to one of the permissions?
- Is the user associated with a role that is directly connected to one of the permissions?
- Is the user directly connected to one of the permissions?

```
permissions.user.check.algorithm=3
```

The fourth algorithm does a database join and checks the permissions that algorithm three checks in one step, by calling `countByGroupsPermissions`, `countByRolesPermissions`, and `countByUsersPermissions` in one method.

```
permissions.user.check.algorithm=4
```

Set the default permissions list filter class. This class must implement `com.liferay.portal.kernel.security.permission.PermissionsListFilter`. This is used if you want to filter the list of permissions before it is actually persisted. For example, if you want to make sure that all users who create objects never have the UPDATE action, then you can filter that list and remove any permissions that have the UPDATE action before it is persisted.

```
permissions.list.filter=com.liferay.portal.security.permission.PermissionsListFilterImpl
```

Captcha

Set the maximum number of captcha checks per portlet session. Set this value to zero to always check. Set this value to a number less than zero to never check. Unauthenticated users will always be checked on every request if captcha checks is enabled.

```
captcha.max.challenges=1
```

Set whether or not to use captcha checks for the following actions.

```
captcha.check.portal.create_account=true
captcha.check.portal.send_password=true
captcha.check.portlet.message_boards.edit_category=false
captcha.check.portlet.message_boards.edit_message=false
```

Startup Events

Input a list of comma delimited class names that extend `com.liferay.portal.struts.SimpleAction`. These classes will run at the specified event.

The following is a global startup event that runs once when the portal initializes.

```
global.startup.events=com.liferay.portal.events.GlobalStartupAction
```

The following is an application startup event that runs once for every web site instance of the portal that initializes.

```
application.startup.events=com.liferay.portal.events.AppStartupAction
#application.startup.events=com.liferay.portal.events.AppStartupAction,com.l
iferay.portal.events.SampleAppStartupAction
```

Shutdown Events

Input a list of comma delimited class names that extend `com.liferay.portal.struts.SimpleAction`. These classes will run at the specified event.

Global shutdown event that runs once when the portal shuts down.

```
global.shutdown.events=com.liferay.portal.events.GlobalShutdownAction
```

Application shutdown event that runs once for every web site instance of the portal that shuts down.

```
application.shutdown.events=com.liferay.portal.events.AppShutdownAction
```

Programmatically kill the Java process on shutdown. This is a workaround for a bug in Tomcat and Linux where the process hangs on forever.

See http://support.liferay.com/browse/LEP-2048 for more information.

```
shutdown.programmatically.exit=false
```

Portal Events

Input a list of comma delimited class names that extend `com.liferay.portal.struts.Action`. These classes will run before or after the specified event.

Servlet service event: The pre-service events have an associated error page and will forward to that page if an exception is thrown during excecution of the events. The pre-service events process before Struts processes the request.

Examples:

```
servlet.service.events.pre=com.liferay.portal.events.ServicePreAction
servlet.service.events.pre=com.liferay.portal.events.LogMemoryUsageAction,co
m.liferay.portal.events.LogThreadCountAction,com.liferay.portal.events.ServicePreAction
servlet.service.events.pre=com.liferay.portal.events.LogSessionIdAction,com.
liferay.portal.events.ServicePreAction
servlet.service.events.pre=com.liferay.portal.events.ServicePreAction,com.li
feray.portal.events.RandomLayoutAction
servlet.service.events.pre=com.liferay.portal.events.ServicePreAction,com.li
feray.portal.events.RandomLookAndFeelAction
```

Use the following to define the error page.

```
servlet.service.events.pre.error.page=/common/error.jsp
```

The post-service events process after Struts processes the request.

```
servlet.service.events.post=com.liferay.portal.events.ServicePostAction
```

Login Event

Define events that can occur pre-login and post-login.

```
login.events.pre=com.liferay.portal.events.LoginPreAction
login.events.post=com.liferay.portal.events.LoginPostAction,com.liferay.portal.
events.DefaultLandingPageAction
```

Logout Event

Similarly, events can be defined for the log out event.

```
logout.events.pre=com.liferay.portal.events.LogoutPreAction
```

Example post events:

```
logout.events.post=com.liferay.portal.events.LogoutPostAction
logout.events.post=com.liferay.portal.events.LogoutPostAction,com.liferay.-
portal.events.GarbageCollectorAction
```

Default Landing Page

Set the default landing page path for logged in users relative to the server path. This is the page users are automatically redirected to after logging in. For example, if you want the default landing page to be `http://localhost:8080/web/guest/login`, set this to `/web/guest/login`. To activate this feature, set `auth.forward.by.last.path` to true. To customize the behavior, see `com.liferay.portal.events.DefaultLandingPageAction` in the `login.events.post` property above.

```
#default.landing.page.path=/web/guest/login
```

Default Logout Page

Set the default logout page path for users relative to the server path. This is the page users are automatically redirected to after logging out. For example, if you want the default logout page to be `http://localhost:8080/web/guest/logout`, set this to `/web/guest/logout`. To activate this feature, set `auth.forward.by.last.path` to true. To customize the behavior, see `com.liferay.portal.events.DefaultLogoutPageAction` in the `logout.events.post` property above.

```
#default.logout.page.path=/web/guest/logout
```

Default Guest Public Layout

The Guest group must have at least one public page. The settings for the initial public page are specified in the following properties.

If you need to add more than one page, set the property `default.guest.public.layout.lar` to specify a LAR file instead.

For even more complex behavior, override the `addDefaultGuestPublicLayouts` method in `com.liferay.portal.service.impl.GroupLocalServiceImpl`.

Set the name of the public layout.

```
default.guest.public.layout.name=Welcome
```

Set the layout template id of the public layout.

```
default.guest.public.layout.template.id=2_columns_ii
```

Set the portlet ids for the columns specified in the layout template.

```
default.guest.public.layout.column-1=58
default.guest.public.layout.column-2=47
default.guest.public.layout.column-3=
default.guest.public.layout.column-4=
```

Set the friendly url of the public layout.

```
default.guest.public.layout.friendly.url=/home
```

Set the regular theme id for the public layout.

```
#default.guest.public.layout.regular.theme.id=classic
```

Set the regular color scheme id for the public layout.

```
#default.guest.public.layout.regular.color.scheme.id=01
```

Set the wap theme id for the public layout.

```
#default.guest.public.layout.wap.theme.id=mobile
```

Set the wap color scheme for the public layout.

```
#default.guest.public.layout.wap.color.scheme.id=01
```

Specify a LAR file that can be used to create the guest public layouts. If this property is set, the previous layout properties will be ignored.

```
#default.guest.public.layouts.lar=${resource.repositories.root}/deploy/default_
guest_public.lar
```

Default User Private Layouts

If the properties `layout.user.private.layouts.enabled` and `layout.user.private.layouts.auto.create` are both set to true, then users will have private layouts and they will be automatically created. The settings below are used for the creation of for the initial private pages.

If you need to add more than one page, set the property `default.user.private.layout.lar` to specify a LAR file instead.

For even more complex behavior, override the `addDefaultUserPrivateLayouts` method in `com.liferay.portal.events.ServicePreAction`.

Set the name of the private layout.

```
default.user.private.layout.name=Welcome
```

Set the layout template id of the private layout.

```
default.user.private.layout.template.id=2_columns_ii
```

Set the portlet ids for the columns specified in the layout template.

```
default.user.private.layout.column-1=71_INSTANCE_OY0d,82,23,61
default.user.private.layout.column-2=11,29,8,19
default.user.private.layout.column-3=
default.user.private.layout.column-4=
```

Set the friendly url of the private layout.

```
default.user.private.layout.friendly.url=/home
```

Set the regular theme id for the private layout.

```
#default.user.private.layout.regular.theme.id=classic
```

Set the regular color scheme id for the private layout.

```
#default.user.private.layout.regular.color.scheme.id=01
```

Set the wap theme id for the private layout.

```
#default.user.private.layout.wap.theme.id=mobile
```

Set the wap color scheme for the private layout.

```
#default.user.private.layout.wap.color.scheme.id=01
```

Specify a LAR file that can be used to create the user private layouts. If this property is set, the previous layout properties will be ignored.

```
#default.user.private.layouts.lar=${resource.repositories.root}/deploy/default_
user_private.lar
```

Default User Public Layouts

If the properties `layout.user.public.layouts.enabled` and `layout.user.public.layouts.auto.create` are both set to true, then users will have public layouts and they will be automatically created. The settings below are used for the creation of for the initial public pages.

If you need to add more than one page, set the property `default.user.public.layout.lar` to specify a LAR file instead.

For even more complex behavior, override the `addDefaultUserPublicLayouts` method in `com.liferay.portal.events.ServicePreAction`.

Set the name of the public layout.

```
default.user.public.layout.name=Welcome
```

Set the layout template id of the public layout.

```
default.user.public.layout.template.id=2_columns_ii
```

Set the portlet ids for the columns specified in the layout template.

```
default.user.public.layout.column-1=82,23
default.user.public.layout.column-2=8,19
default.user.public.layout.column-3=
default.user.public.layout.column-4=
```

Set the friendly url of the public layout.

```
default.user.public.layout.friendly.url=/home
```

Set the regular theme id for the public layout.

```
#default.user.public.layout.regular.theme.id=classic
```

Set the regular color scheme id for the public layout.

```
#default.user.public.layout.regular.color.scheme.id=01
```

Set the wap theme id for the public layout.

```
#default.user.public.layout.wap.theme.id=mobile
```

Set the wap color scheme for the public layout.

```
#default.user.public.layout.wap.color.scheme.id=01
```

Specify a LAR file that can be used to create the user public layouts. If this property is set, the previous layout properties will be ignored.

```
#default.user.public.layouts.lar=${resource.repositories.root}/deploy/default_
user_public.lar
```

Default Admin

Set the default admin password.

```
default.admin.password=test
```

Set the default admin screen name prefix.

```
default.admin.screen.name=test
```

Set the default admin email address prefix.

```
default.admin.email.address.prefix=test
```

Set the default admin first name.

```
default.admin.first.name=Test
```

Set the default admin middle name.

```
default.admin.middle.name=
```

Set the default admin last name.

```
default.admin.last.name=Test
```

Layouts

Set the list of layout types. The display text of each of the layout types is set in *content/Language.properties* and prefixed with *layout.types*. You can create new layout types and specify custom settings for each layout type. End users input dynamic values as designed in the edit page. End users see the layout as designed in the view page. The generated URL can reference properties set in the edit page. Parentable layouts can contain child layouts. You can also specify a comma delimited list of configuration actions that will be called for your layout when it is updated or deleted.

```
layout.types=portlet,panel,embedded,article,url,link_to_layout
```

Set whether or not private layouts are enabled. Set whether or not private layouts are modifiable. Set whether or not private layouts should be auto created if a user has no private layouts. If private layouts are not enabled, the other two properties are assumed to be false.

```
layout.user.private.layouts.enabled=true
layout.user.private.layouts.modifiable=true
layout.user.private.layouts.auto.create=true
```

Set whether or not public layouts are enabled. Set whether or not public layouts are modifiable. Set whether or not public layouts should be auto created if a user has no public layouts. If public layouts are not enabled, the other two properties are assumed to be false.

```
layout.user.public.layouts.enabled=true
layout.user.public.layouts.modifiable=true
layout.user.public.layouts.auto.create=true
```

Default Settings Layout

```
layout.edit.page=/portal/layout/edit/portlet.jsp
layout.view.page=/portal/layout/view/portlet.jsp
layout.url=${liferay:mainPath}/portal/layout?p_l_id=${liferay:plid}
layout.url.friendliable=true
layout.parentable=true
layout.sitemapable=true
layout.configuration.action.update=
layout.configuration.action.delete=
```

Settings for portlet layouts are inherited from the default settings.

```
layout.edit.page[portlet]=/portal/layout/edit/portlet.jsp
layout.view.page[portlet]=/portal/layout/view/portlet.jsp
layout.url[portlet]=${liferay:mainPath}/portal/layout?p_l_id=${liferay:plid}
layout.url.friendliable[portlet]=true
layout.parentable[portlet]=true
layout.configuration.action.update[portlet]=
layout.configuration.action.delete[portlet]=
```

Settings for panel layouts.

```
layout.edit.page[panel]=/portal/layout/edit/panel.jsp
layout.view.page[panel]=/portal/layout/view/panel.jsp
layout.url[panel]=${liferay:mainPath}/portal/layout?p_l_id=${liferay:plid}
layout.url.friendliable[panel]=true
layout.parentable[panel]=true
```

Settings for embedded layouts.

```
layout.edit.page[embedded]=/portal/layout/edit/embedded.jsp
layout.view.page[embedded]=/portal/layout/view/embedded.jsp
layout.url[embedded]=${liferay:mainPath}/portal/layout?p_l_id=${liferay:
plid}
layout.url.friendliable[embedded]=true
layout.parentable[embedded]=false
layout.sitemapable[embedded]=true
layout.configuration.action.update[embedded]=
layout.configuration.action.delete[embedded]=
```

Settings for article layouts.

```
layout.edit.page[article]=/portal/layout/edit/article.jsp
layout.view.page[article]=/portal/layout/view/article.jsp
layout.url.friendliable[article]=true
layout.url[article]=${liferay:mainPath}/portal/layout?p_l_id=${liferay:plid}
layout.parentable[article]=false
layout.sitemapable[article]=true
layout.configuration.action.update[article]=com.liferay.portal.model.Layout-
TypeArticleConfigurationUpdateAction
layout.configuration.action.delete[article]=com.liferay.portal.model.Layout-
TypeArticleConfigurationDeleteAction
```

Settings for URL layouts.

```
layout.edit.page[url]=/portal/layout/edit/url.jsp
layout.view.page[url]=
layout.url[url]=${url}
layout.url.friendliable[url]=true
layout.parentable[url]=false
layout.sitemapable[url]=false
layout.configuration.action.update[url]=
layout.configuration.action.delete[url]=
```

Settings for page layouts.

```
layout.edit.page[link_to_layout]=/portal/layout/edit/link_to_layout.jsp
layout.view.page[link_to_layout]=
layout.url[link_to_layout]=${liferay:mainPath}/portal/layout?p_l_id=${link-
ToPlid}
layout.url.friendliable[link_to_layout]=true
layout.parentable[link_to_layout]=true
layout.sitemapable[link_to_layout]=false
layout.configuration.action.update[link_to_layout]=
layout.configuration.action.delete[link_to_layout]=
```

Specify static portlets that cannot be moved and will always appear on every layout. Static portlets will take precedence over portlets that may have been dynamically configured for the layout.

For example, if you want the Hello World portlet to always appear at the start of the iteration of the first column for user layouts, set the property `layout.static.portlets.start.column-1[user]` to 47. If you want the Hello World portlet to always appear at the end of the second column for user layouts, set the property `layout.static.portlets.end.column-2[user]` to 47. You can input a list of comma delimited portlet ids to specify more than one portlet. If the portlet is instanceable, add the *suffix _INSTANCE_abcd* to the portlet id, where abcd is any random alphanumeric string.

The static portlets are fetched based on the properties controlled by custom filters using EasyConf. By default, the available filters are user, community, and organization.

```
layout.static.portlets.start.column-1[user]=3,6
layout.static.portlets.end.column-1[user]=14
layout.static.portlets.start.column-2[user]=71_INSTANCE_abcd,7
layout.static.portlets.end.column-2[user]=34,70
layout.static.portlets.start.column-3[user]=
layout.static.portlets.end.column-3[user]=
```

It is also possible to set static portlets based on the layout's friendly URL.

```
layout.static.portlets.start.column-1[user][/home]=3,6
layout.static.portlets.end.column-2[community][/home]=14
```

Set the static layouts for community layouts.

```
layout.static.portlets.start.column-1[community]=
layout.static.portlets.end.column-1[community]=
layout.static.portlets.start.column-2[community]=
layout.static.portlets.end.column-2[community]=
layout.static.portlets.start.column-3[community]=
layout.static.portlets.end.column-3[community]=
```

Set the static layouts for organization layouts.

```
layout.static.portlets.start.column-1[organization]=
layout.static.portlets.end.column-1[organization]=
layout.static.portlets.start.column-2[organization]=
layout.static.portlets.end.column-2[organization]=
layout.static.portlets.start.column-3[organization]=
layout.static.portlets.end.column-3[organization]=
```

Set the private group, private user, and public servlet mapping for `com.liferay.portal.servlet.FriendlyURLServlet`. This value must match the servlet mapping set in `web.xml`.

For example, if the private group pages are mapped to */group* and the group's friendly URL is set to */guest* and the layout's friendly URL is set to */company/community*, then the friendly URL for the page will be *http://www.liferay.com/group/guest/company/community*. Private group pages map to a community's private pages and are only available to authenticated users with the proper permissions.

For example, if the public pages are mapped to */web* and the group or user's friendly URL is set to */guest* and the layout's friendly URL is set to */company/community*, then the friendly URL for the page will be *http://www.liferay.com/web/guest/company/ community*. Public pages are available to unauthenticated users.

The friendly URLs for users, groups, and layouts can be set during runtime.

```
layout.friendly.url.private.group.servlet.mapping=/group
layout.friendly.url.private.user.servlet.mapping=/user
layout.friendly.url.public.servlet.mapping=/web
```

Redirect to this resource if the user requested a friendly URL that does not exist. Leave it blank to display nothing.

Note – For backward compatibility, this overrides the property layout.show.http.status for the 404 status code.

```
layout.friendly.url.page.not.found=/html/portal/404.html
```

Set the reserved keywords that cannot be used in a friendly URL.

```
layout.friendly.url.keywords=c,group,web,image,wsrp,page,public,private,rss,
tags
```

Set the following to true if layouts should remember (across requests) that a window state was set to maximized.

```
layout.remember.request.window.state.maximized=false
```

Set the following to true if guest users should see the maximize window icon.

```
layout.guest.show.max.icon=false
```

Set the following to true if guest users should see the minimize window icon.

```
layout.guest.show.min.icon=false
```

Set the following to true if users are shown that they do not have access to a portlet. The portlet init parameter show-portlet-access-denied will override this setting.

```
layout.show.portlet.access.denied=true
```

Set the following to true if users are shown that a portlet is inactive. The portlet init parameter show-portlet-inactive will override this setting.

```
layout.show.portlet.inactive=true
```

Set the following to true if the portal should show HTTP status codes like 404 if the requested page is not found.

```
layout.show.http.status=true
```

Set the default layout template id used when creating layouts.

```
layout.default.template.id=2_columns_ii
```

Set the following to false to disable parallel rendering. You can also disable it on a per request basis by setting the attribute key `com.liferay.portal.util.WebKeys.PORTLET_PARALLEL_RENDER` to the `Boolean.FALSE` in a pre service event or by setting the URL parameter `p_p_parallel` to 0.

```
layout.parallel.render.enable=true
```

Set the name of a class that implements `com.liferay.portal.util.LayoutClone`. This class is used to remember maximized and minimized states on shared pages. The default implementation persists the state in the browser session.

```
layout.clone.impl=com.liferay.portal.util.SessionLayoutClone
```

Set the following to true to cache the content of layout templates. This is recommended because it improves performance for production servers. Setting it to false is useful during development if you need to make a lot of changes.

```
layout.template.cache.enabled=true
```

Set the default value for the *p_l_reset* parameter. If set to true, then render parameters are cleared when different pages are hit. This is not the behavior promoted by the portlet specification, but is the one that most end users seem to prefer.

```
layout.default.p_l_reset=true
```

Portlet URL

Set the following to true if calling setParameter on a portlet URL appends the parameter value versus replacing it. There is some disagreement in the interpretation of the JSR 168 spec among portlet developers over this specific behavior. Liferay Portal successfully passes the portlet TCK tests whether this value is set to true or false.

See http://support.liferay.com/browse/LEP-426 for more information.

```
portlet.url.append.parameters=false
```

Set the following to true to allow portlet URLs to generate with an anchor tag.

```
portlet.url.anchor.enable=false
```

JSR 286 specifies that portlet URLs are escaped by default. Set this to false to provide for better backwards compatibility.

If this is set to true, but a specific portlet application requires that its portlet URLs not be escaped by default, then modify `portlet.xml` and set the container run-time option `javax.portlet.escapeXml` to false.

```
portlet.url.escape.xml=false
```

Preferences

Set the following to true to validate portlet preferences on startup.

```
preference.validate.on.startup=false
```

Struts

Input the custom Struts request processor that will be used by Struts based portlets. The custom class must extend `com.liferay.portal.struts.PortletRequestProcessor` and have the same constructor.

```
struts.portlet.request.processor=com.liferay.portal.struts.PortletRequest-
Processor
```

Images

Set the location of the default spacer image that is used for missing images. This image must be found in the class path.

```
image.default.spacer=com/liferay/portal/dependencies/spacer.gif
```

Set the location of the default company logo image that is used for missing company logo images. This image must be found in the class path.

```
image.default.company.logo=com/liferay/portal/dependencies/company_logo.png
```

Set the locations of the default user portrait images that are used for missing user portrait images. This image must be found in the class path.

```
image.default.user.female.portrait=com/liferay/portal/dependencies/user_female_
portrait.gif
image.default.user.male.portrait=com/liferay/portal/dependencies/user_male_p
ortrait.gif
```

Editors

You can configure individual JSP pages to use a specific implementation of the available WYSIWYG editors: *liferay*, *fckeditor*, *simple*, *tinymce*, or *tinymcesimple*.

```
editor.wysiwyg.default=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.blogs.edit_entry.jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.calendar.edit_configuration.
jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.enterprise_admin.
view.jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.invitation.edit_configuration.
jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.journal.edit_article_content.
jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.journal.
edit_article_content_xsd_el.jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.journal.edit_configuration.
jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.mail.edit.jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.mail.edit_message.jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.message_boards.edit_configuration.
jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.shopping.edit_configuration.
jsp=fckeditor
editor.wysiwyg.portal-web.docroot.html.portlet.wiki.edit_html.jsp=fckeditor
```

Fields

Set the following fields to false so users cannot see them. Some company policies require gender and birthday information to always be hidden.

```
field.enable.com.liferay.portal.model.Contact.male=true
field.enable.com.liferay.portal.model.Contact.birthday=true
field.enable.com.liferay.portal.model.Organization.status=false
```

MIME Types

Input a list of comma delimited mime types that are not available by default from `javax.activation.MimetypesFileTypeMap`.

```
mime.types=\
application/pdf pdf,\
application/vnd.ms-excel xls,\
application/vnd.ms-powerpoint ppt,\
application/msword doc
```

Amazon Licence Keys

Enter a list of valid Amazon license keys. Configure additional keys by incrementing the last number. The keys are used following a Round-Robin algorithm. This is made available only for personal use. Please see the Amazon license at http://www.amazon.com for more information.

```
amazon.license.0=
amazon.license.1=
amazon.license.2=
amazon.license.3=
```

Instant Messenger

Set the AIM login and password which the system will use to communicate with users.

```
aim.login=
aim.password=
```

Due to a bug in *JOscarLib 0.3b1*, you must set the full path to the ICQ jar.

See the following posts:

`http://sourceforge.net/forum/message.php?msg_id=1972697`

`http://sourceforge.net/forum/message.php?msg_id=1990487`

```
icq.jar=C:/Java/orion-2.0.7/lib/icq.jar
```

Set the ICQ login and password which the system will use to communicate with users.

```
icq.login=
icq.password=
```

Set the MSN login and password which the system will use to communicate with users.

```
msn.login=
msn.password=
```

Set the YM login and password which the system will use to communicate with users.

```
ym.login=
ym.password=
```

Lucene Search

Set the following to true if you want to avoid any writes to the index. This is useful in some clustering environments where there is a shared index and only one node of the cluster updates it.

```
index.read.only=false
```

Set the following to true if you want to index your entire library of files on startup.

```
index.on.startup=false
```

Set the following to true if you want the indexing on startup to be executed on a separate thread to speed up execution.

```
index.with.thread=true
```

Designate whether Lucene stores indexes in a database via JDBC, file system, or in RAM.

Examples:

```
lucene.store.type=jdbc
lucene.store.type=file
lucene.store.type=ram
```

Lucene's storage of indexes via JDBC has a bug where temp files are not removed. This can eat up disk space over time. Set the following property to true to automatically clean up the temporary files once a day. See LEP-2180.

```
lucene.store.jdbc.auto.clean.up=true
```

Set the JDBC dialect that Lucene uses to store indexes in the database. This is only referenced if Lucene stores indexes in the database. Liferay will attempt to load the proper dialect based on the URL of the JDBC connection. For example, the property `lucene.store.jdbc.dialect.mysql` is read for the JDBC connection URL `jdbc:mysql://localhost/lportal`.

```
lucene.store.jdbc.dialect.db2=org.apache.lucene.store.jdbc.dialect.DB2Dialect
lucene.store.jdbc.dialect.derby=org.apache.lucene.store.jdbc.dialect.Derby-
Dialect
lucene.store.jdbc.dialect.hsqldb=org.apache.lucene.store.jdbc.dialect.HSQLDialect
lucene.store.jdbc.dialect.jtds=org.apache.lucene.store.jdbc.dialect.
SQLServerDialect
lucene.store.jdbc.dialect.microsoft=org.apache.lucene.store.jdbc.dialect.
SQLServerDialect
lucene.store.jdbc.dialect.mysql=org.apache.lucene.store.jdbc.dialect.MySQLDialect
#lucene.store.jdbc.dialect.mysql=org.apache.lucene.store.jdbc.dialect.
MySQLInnoDBDialect
#lucene.store.jdbc.dialect.mysql=org.apache.lucene.store.jdbc.dialect.
MySQLMyISAMDialect
lucene.store.jdbc.dialect.oracle=org.apache.lucene.store.jdbc.dialect.OracleDialect
lucene.store.jdbc.dialect.postgresql=org.apache.lucene.store.jdbc.dialect.-
PostgreSQLDialect
```

Set the directory where Lucene indexes are stored. This is only referenced if Lucene stores indexes in the file system.

```
lucene.dir=${resource.repositories.root}/lucene/
```

Input a class name that extends `com.liferay.portal.search.lucene.LuceneFileExtractor`. This class is called by Lucene to extract text from complex files so that they can be properly indexed.

```
lucene.file.extractor=com.liferay.portal.search.lucene.LuceneFileExtractor
```

The file extractor can sometimes return text that is not valid for Lucene. This property expects a regular expression. Any character that does not match the regular expression will be replaced with a blank space. Set an empty regular expression to disable this feature.

Examples:

```
lucene.file.extractor.regexp.strip=
lucene.file.extractor.regexp.strip=[\\d\\w]
```

Set the default analyzer used for indexing and retrieval.

Examples:

```
lucene.analyzer=org.apache.lucene.analysis.br.BrazilianAnalyzer
lucene.analyzer=org.apache.lucene.analysis.cn.ChineseAnalyzer
lucene.analyzer=org.apache.lucene.analysis.cjk.CJKAnalyzer
lucene.analyzer=org.apache.lucene.analysis.cz.CzechAnalyzer
lucene.analyzer=org.apache.lucene.analysis.nl.DutchAnalyzer
lucene.analyzer=org.apache.lucene.analysis.fr.FrenchAnalyzer
lucene.analyzer=org.apache.lucene.analysis.de.GermanAnalyzer
lucene.analyzer=org.apache.lucene.analysis.KeywordAnalyzer
lucene.analyzer=org.apache.lucene.index.memory.PatternAnalyzer
lucene.analyzer=org.apache.lucene.analysis.PerFieldAnalyzerWrapper
lucene.analyzer=org.apache.lucene.analysis.ru.RussianAnalyzer
lucene.analyzer=org.apache.lucene.analysis.SimpleAnalyzer
lucene.analyzer=org.apache.lucene.analysis.snowball.SnowballAnalyzer
lucene.analyzer=org.apache.lucene.analysis.standard.StandardAnalyzer
lucene.analyzer=org.apache.lucene.analysis.StopAnalyzer
lucene.analyzer=org.apache.lucene.analysis.WhitespaceAnalyzer
```

Set Lucene's merge factor. Higher numbers mean indexing goes faster but uses more memory. The default value from Lucene is 10. This should never be set to a number lower than 2.

```
lucene.merge.factor=10
```

Set how often to run Lucene's optimize method. Optimization speeds up searching but slows down writing. Set this property to 0 to always optimize. Set this property to an integer greater than 0 to optimize every X writes.

```
lucene.optimize.interval=1
```

SourceForge

```
source.forge.mirrors=\
http://downloads.sourceforge.net,\ # Redirect
http://internap.dl.sourceforge.net,\ # San Jose, CA
http://superb-east.dl.sourceforge.net,\ # McLean, Virginia
http://superb-west.dl.sourceforge.net,\ # Seattle, Washington
http://easynews.dl.sourceforge.net,\ # Phoenix, AZ
http://kent.dl.sourceforge.net,\ # Kent, UK
http://ufpr.dl.sourceforge.net,\ # Curitiba, Brazil
http://belnet.dl.sourceforge.net,\ # Brussels, Belgium
http://switch.dl.sourceforge.net,\ # Lausanne, Switzerland
http://mesh.dl.sourceforge.net,\ # Duesseldorf, Germany
http://ovh.dl.sourceforge.net,\ # Paris, France
http://dfn.dl.sourceforge.net,\ # Berlin, Germany
http://heanet.dl.sourceforge.net,\ # Dublin, Ireland
http://garr.dl.sourceforge.net,\ # Bologna, Italy
http://surfnet.dl.sourceforge.net # Amsterdam, The Netherlands
http://jaist.dl.sourceforge.net,\ # Ishikawa, Japan
http://nchc.dl.sourceforge.net,\ # Tainan, Taiwan
http://optusnet.dl.sourceforge.net # Sydney, Australia
```

Value Object

You can add a listener for a specific class by setting the property `value.object.listener` with a list of comma delimited class names that implement `com.liferay.portal.model.ModelListener`. These classes are pooled and reused and must be thread safe.

```
value.object.listener.com.liferay.portal.model.Contact=com.liferay.portal.-
model.ContactListener
value.object.listener.com.liferay.portal.model.Layout=com.liferay.portal.-
model.LayoutListener
value.object.listener.com.liferay.portal.model.LayoutSet=com.liferay.portal.
model.LayoutSetListener
value.object.listener.com.liferay.portal.model.PortletPreferences=com.liferay.
portal.model.PortletPreferencesListener
value.object.listener.com.liferay.portal.model.User=com.liferay.portal.model.
UserListener
value.object.listener.com.liferay.portlet.journal.model.JournalArticle=com.l
iferay.portlet.journal.model.JournalArticleListener
value.object.listener.com.liferay.portlet.journal.model.JournalTemplate=com.
liferay.portlet.journal.model.JournalTemplateListener
```

Value objects are cached by default. You can disable caching for all objects or per object.

For mapping tables, the key is the mapping table itself.

```
value.object.finder.cache.enabled=true
value.object.finder.cache.enabled.com.liferay.portal.model.Layout=true
value.object.finder.cache.enabled.com.liferay.portal.model.User=true
value.object.finder.cache.enabled.Users_Roles=true
```

Last Modified

Set the following to true to check last modified date on server side CSS and JavaScript.

```
last.modified.check=true
```

Enter a list of comma delimited paths that will only be executed when newer than the last modified date. These paths must extend `com.liferay.portal.lastmodified.LastModifiedAction`.

```
last.modified.paths=\
/portal/css_cached,\
/portal/javascript_cached
```

XSS (Cross Site Scripting)

Set the following to false to ensure that all persisted data is stripped of XSS hacks.

```
xss.allow=false
```

You can override the *xss.allow* setting for a specific class by setting the property *xss.allow* plus the class name.

```
xss.allow.com.liferay.portal.model.Portlet=true
xss.allow.com.liferay.portal.model.PortletPreferences=true
```

You can override the xss.allow setting for a specific field in a class by setting the property xss.allow plus the class and field name.

```
xss.allow.com.liferay.portlet.journal.model.JournalArticle.content=true
xss.allow.com.liferay.portlet.journal.model.JournalStructure.xsd=true
xss.allow.com.liferay.portlet.journal.model.JournalTemplate.xsl=true
```

Communication Link

Set the JGroups properties used by the portal to communicate with other instances of the portal. This is only needed if the portal is running in a clustered environment. The JGroups settings provide a mechanism for the portal to broadcast messages to the other instances of the portal. The specified multi-cast address should be unique for internal portal messaging only. You will still need to set the Hibernate and Ehcache settings for database clustering.

```
comm.link.properties=UDP(bind_addr=127.0.0.1;mcast_addr=231.12.21.102;mcast_
port=45566;ip_ttl=32;mcast_send_buf_size=150000;mcast_recv_buf_size=80000):P
ING(timeout=2000;num_initial_members=3):MERGE2(min_interval=5000;max_interval=
10000):FD_SOCK:VERIFY_SUSPECT(timeout=1500):pbcast.NAKACK(gc_lag=50;retransmit_
timeout=300,600,1200,2400,4800;max_xmit_size=8192):UNICAST(timeout=
300,600,1200,2400):pbcast.STABLE(desired_avg_gossip=20000):FRAG(frag_size=80
96;down_thread=false;up_thread=false):pbcast.GMS(join_timeout=5000;join_retr
y_timeout=2000;shun=false;print_local_addr=true)
```

Content Delivery Network

Set the hostname that will be used to serve static content via a CDN. This property can be overridden dynamically at runtime by setting the HTTP parameter *cdn_host*.

```
cdn.host=
```

Counter

Set the number of increments between database updates to the Counter table. Set this value to a higher number for better performance.

```
counter.increment=100
```

Lock

Set the lock expiration time for each class.

Example: 1 Day

```
lock.expiration.time.com.liferay.portlet.documentlibrary.model.DLFileEntry=8
6400000
```

Example: 20 Minutes

```
lock.expiration.time.com.liferay.portlet.wiki.model.WikiPage=1200000
```

JBI

Connect to either Mule or ServiceMix as your ESB.

Examples:

```
jbi.workflow.url=http://localhost:8080/mule-web/workflow
jbi.workflow.url=http://localhost:8080/servicemix-web/workflow
```

JCR

Liferay includes Jackrabbit (http://jackrabbit.apache.org) by default as its JSR-170 Java Content Repository.

```
jcr.initialize.on.startup=false
jcr.workspace.name=liferay
jcr.node.documentlibrary=documentlibrary
jcr.jackrabbit.repository.root=${resource.repositories.root}/jackrabbit
jcr.jackrabbit.config.file.path=${jcr.jackrabbit.repository.root}/repository.
xml
jcr.jackrabbit.repository.home=${jcr.jackrabbit.repository.root}/home
jcr.jackrabbit.credentials.username=none
jcr.jackrabbit.credentials.password=none
```

OpenOffice

Enabling OpenOffice integration allows the Document Library portlet to provide document conversion functionality. To start OpenOffice as a service, run the command:

```
soffice -headless -accept="socket,host=127.0.0.1,port=8100;urp;" -nofirststartwizard
```

This is tested with OpenOffice 2.3.x.

```
openoffice.server.enabled=false
openoffice.server.host=127.0.0.1
openoffice.server.port=8100
```

POP

Set this to true to enable polling of email notifications from a POP server. The user credentials are the same used for SMTP authentication and is specified in the `mail/MailSession` configuration for each application server.

```
pop.server.notifications.enabled=false
```

Set the interval on which the POPNotificationsJob will run. The value is set in one minute increments.

```
pop.server.notifications.interval=1
pop.server.subdomain=events
```

Quartz

These properties define the connection to the built-in Quartz job scheduling engine.

```
org.quartz.dataSource.ds.connectionProvider.class=com.liferay.portal.scheduler.
quartz.QuartzConnectionProviderImpl
org.quartz.jobStore.class=org.quartz.impl.jdbcjobstore.JobStoreTX
org.quartz.jobStore.dataSource=ds
org.quartz.jobStore.driverDelegateClass=com.liferay.portal.scheduler.
quartz.DynamicDriverDelegate
org.quartz.jobStore.isClustered=false
org.quartz.jobStore.misfireThreshold=60000
org.quartz.jobStore.tablePrefix=QUARTZ_
org.quartz.jobStore.useProperties=true
org.quartz.scheduler.instanceId=AUTO
org.quartz.threadPool.class=org.quartz.simpl.SimpleThreadPool
org.quartz.threadPool.threadCount=5
org.quartz.threadPool.threadPriority=5
```

Scheduler

Set this to false to disable all scheduler classes defined in `liferay-portlet.xml` and in the property `scheduler.classes`.

```
scheduler.enabled=true
```

Input a list of comma delimited class names that implement `com.liferay.portal.kernel.job.Scheduler`. These classes allow jobs to be scheduled on startup. These classes are not associated to any one portlet.

```
scheduler.classes=
```

Social Bookmarks

The Blogs portlet allows for the posting of entries to various popular social bookmarking sites. The example ones are the defaults; to configure more, just add the site in the format below.

```
social.bookmark.types=blinklist,delicious,digg,furl,newsvine,reddit,technorati
social.bookmark.post.url[blinklist]=http://blinklist.com/index.php?
Action=Blink/addblink.php&url=${liferay:social-bookmark:url}&Title=${liferay:
social-bookmark:title}
social.bookmark.post.url[delicious]=http://del.icio.us/post?url=${liferay:
social-bookmark:url}&title=${liferay:social-bookmark:title}
social.bookmark.post.url[digg]=http://digg.com/submit?phase=2&url=${liferay:
social-bookmark:url}
social.bookmark.post.url[furl]=http://furl.net/storeIt.jsp?u=${liferay:social-
bookmark:url}&t=${liferay:social-bookmark:title}
social.bookmark.post.url[newsvine]=http://www.newsvine.com/_tools/seed&save?
u=${liferay:social-bookmark:url}&h=${liferay:social-bookmark:title}
social.bookmark.post.url[reddit]=http://reddit.com/submit?url=${liferay:social-
bookmark:url}&title=${liferay:social-bookmark:title}
social.bookmark.post.url[technorati]=http://technorati.com/cosmos/search.htm
l?url=${liferay:social-bookmark:url}
```

Velocity Engine

Input a list of comma delimited class names that extend `com.liferay.util.velocity.VelocityResourceListener`. These classes will run in sequence to allow you to find the applicable ResourceLoader to load a Velocity template.

```
velocity.engine.resource.listeners=com.liferay.portal.velocity.ServletVelocityResourceListener,
com.liferay.portal.velocity.JournalTemplateVelocityResourceListener,
com.liferay.portal.velocity.ThemeLoaderVelocityResourceListener,
com.liferay.portal.velocity.ClassLoaderVelocityResourceListener
```

Set the Velocity resource managers. We extend the Velocity's default resource managers for better scalability.

Note that the modification check interval is not respected because the resource loader implementation does not know the last modified date of a resource. This means you will need to turn off caching if you want to be able to modify VM templates in themes and see the changes right away.

```
velocity.engine.resource.manager=com.liferay.portal.velocity.LiferayResourceManager
velocity.engine.resource.manager.cache=com.liferay.portal.velocity.LiferayResourceCache
velocity.engine.resource.manager.cache.enabled=true
#velocity.engine.resource.manager.modification.check.interval=0
```

Input a list of comma delimited macros that will be loaded. These files must exist in the class path.

```
velocity.engine.velocimacro.library=VM_global_library.vm,VM_liferay.vm
```

Set the Velocity logging configuration.

```
velocity.engine.logger=org.apache.velocity.runtime.log.SimpleLog4JLogSystem
velocity.engine.logger.category=org.apache.velocity
```

Virtual Hosts

Set the hosts that will be ignored for virtual hosts.

```
virtual.hosts.ignore.hosts=\
127.0.0.1,\
localhost
```

Set the paths that will be ignored for virtual hosts.

```
virtual.hosts.ignore.paths=\
/c,\
\
/c/portal/change_password,\
/c/portal/css_cached,\
/c/portal/extend_session,\
/c/portal/extend_session_confirm,\
/c/portal/javascript_cached,\
/c/portal/json_service,\
/c/portal/layout,\
/c/portal/login,\
/c/portal/logout,\
/c/portal/render_portlet,\
/c/portal/reverse_ajax,\
/c/portal/session_tree_js_click,\
/c/portal/status,\
/c/portal/update_layout,\
/c/portal/update_terms_of_use,\
/c/portal/upload_progress_poller,\
\
/c/layout_configuration/templates,\
/c/layout_management/update_page
```

HTTP

See `system.properties` for more HTTP settings.

Set the maximum number of connections.

```
#com.liferay.portal.util.HttpImpl.max.connections.per.host=2
#com.liferay.portal.util.HttpImpl.max.total.connections=20
```

Set the proxy authentication type.

```
#com.liferay.portal.util.HttpImpl.proxy.auth.type=username-password
#com.liferay.portal.util.HttpImpl.proxy.auth.type=ntlm
```

Set user name and password used for HTTP proxy authentication.

```
#com.liferay.portal.util.HttpImpl.proxy.username=
#com.liferay.portal.util.HttpImpl.proxy.password=
```

Set additional properties for NTLM authentication.

```
#com.liferay.portal.util.HttpImpl.proxy.ntlm.domain=
#com.liferay.portal.util.HttpImpl.proxy.ntlm.host=
```

Set the connection timeout when fetching HTTP content.

```
com.liferay.portal.util.HttpImpl.timeout=10000
```

Servlet Filters

If the user can unzip compressed HTTP content, the compression filter will zip up the HTTP content before sending it to the user. This will speed up page rendering for users that are on dial up.

```
com.liferay.portal.servlet.filters.compression.CompressionFilter=true
```

This double click filter will prevent double clicks at the server side. Prevention of double clicks is already in place on the client side. However, some sites require a more robust solution. This is turned off by default since most sites will not need it.

```
com.liferay.portal.servlet.filters.doubleclick.DoubleClickFilter=false
```

The header filter is used to set request headers.

```
com.liferay.portal.servlet.filters.header.HeaderFilter=true
```

The strip filter will remove blank lines from the outputted content. This will speed up page rendering for users that are on dial up.

```
com.liferay.portal.servlet.filters.strip.StripFilter=true
```

The layout cache filter will cache pages to speed up page rendering for guest users. See `ehcache.xml` to modify the cache expiration time to live.

```
com.liferay.portal.servlet.filters.layoutcache.LayoutCacheFilter=true
```

The session id filter ensure that only one session is created between http and https sessions. This is useful if you want users to login via https but have them view the rest of the site via http. This is disabled by default. Do not enable this unless you thoroughly understand how cookies, http, and https work.

```
com.liferay.portal.servlet.filters.sessionid.SessionIdFilter=false
```

The Velocity filter will process */css/main.css* as a Velocity template.

```
com.liferay.portal.servlet.filters.velocity.VelocityFilter=false
```

The virtual host filter maps hosts to public and private pages. For example, if the public virtual host is *www.helloworld.com* and the friendly URL is */helloworld*, then *http://www.helloworld.com* is mapped to *http://localhost:8080/web/helloworld*.

```
com.liferay.portal.servlet.filters.virtualhost.VirtualHostFilter=true
```

Upload Servlet Request

Set the maximum file size. Default is 1024 * 1024 * 100.

```
com.liferay.portal.upload.UploadServletRequestImpl.max.size=104857600
```

Set the temp directory for uploaded files.

```
#com.liferay.portal.upload.UploadServletRequestImpl.temp.dir=C:/Temp
```

Set the threshold size to prevent extraneous serialization of uploaded data.

```
com.liferay.portal.upload.LiferayFileItem.threshold.size=262144
```

Set the threshold size to prevent out of memory exceptions caused by caching excessively large uploaded data. Default is 1024 * 1024 * 10.

```
com.liferay.portal.upload.LiferayInputStream.threshold.size=10485760
```

Web Server

Set the HTTP and HTTPs ports when running the portal in a J2EE server that is sitting behind another web server like Apache. Set the values to -1 if the portal is not running behind another web server like Apache.

```
web.server.http.port=-1
web.server.https.port=-1
```

Set the hostname that will be used when the portlet generates URLs. Leaving this blank will mean the host is derived from the servlet container.

```
web.server.host=
```

Set the preferred protocol.

```
web.server.protocol=https
```

Set this to true to display the server name at the bottom of every page. This is useful when testing clustering configurations so that you can know which node you are accessing.

```
web.server.display.node=false
```

WebDAV

Set the following to true to enable programmatic configuration to let the Web- DAV be configured for litmus testing. This should never be set to true unless you are running the litmus tests.

```
webdav.litmus=false
```

Set a list of files for the WebDAV servlet to ignore processing.

```
webdav.ignore=.DS_Store,.metadata_index_homes_only,.metadata_never_index,.Sp
otlight-V100,.TemporaryItems,.Trashes
```

Set the tokens for supported WebDAV storage paths.

```
webdav.storage.tokens=document_library,image_gallery,journal
```

Set the class names for supported WebDAV storage classes.

```
webdav.storage.class[document_library]=com.liferay.portlet.documentlibrary.
webdav.DLWebDAVStorageImpl
webdav.storage.class[image_gallery]=com.liferay.portlet.imagegallery.webdav.
IGWebDAVStorageImpl
webdav.storage.class[journal]=com.liferay.portlet.journal.webdav.JournalWeb-
DAVStorageImpl
```

Main Servlet

Servlets can be protected by *com.liferay.portal.servlet.filters.secure.SecureFilter*.

Input a list of comma delimited IPs that can access this servlet. Input a blank list to allow any IP to access this servlet. SERVER_IP will be replaced with the IP of the host server.

```
main.servlet.hosts.allowed=
```

Set the following to true if this servlet can only be accessed via https.

```
main.servlet.https.required=false
```

Axis Servlet

See Main Servlet on how to protect this servlet.

```
axis.servlet.hosts.allowed=127.0.0.1,SERVER_IP
axis.servlet.https.required=false
```

JSON Tunnel Servlet

See Main Servlet on how to protect this servlet.

```
json.servlet.hosts.allowed=
json.servlet.https.required=false
```

Liferay Tunnel Servlet

See Main Servlet on how to protect this servlet.

```
tunnel.servlet.hosts.allowed=127.0.0.1,SERVER_IP
tunnel.servlet.https.required=false
```

Spring Remoting Servlet

See Main Servlet on how to protect this servlet.

```
spring.remoting.servlet.hosts.allowed=127.0.0.1,SERVER_IP
spring.remoting.servlet.https.required=false
```

WebDAV Servlet

See Main Servlet on how to protect this servlet.

```
webdav.servlet.hosts.allowed=
webdav.servlet.https.required=false
```

Admin Portlet

You can set some administrative defaults by using these properties. The first time you bring up your portal, these values will then already be set in the Admin portlet. All values should be separated by \n characters.

Set up default group names.

```
admin.default.group.names=
```

Set up default role names.

```
admin.default.role.names=Power User\nUser
```

Set up default user group names.

```
admin.default.user.group.names=
```

The rest of these properties map to their values in the Admin portlet.

```
admin.mail.host.names=
admin.reserved.screen.names=
admin.reserved.email.addresses=
admin.email.from.name=Joe Bloggs
admin.email.from.address=test@liferay.com
admin.email.user.added.enabled=true
admin.email.user.added.subject=com/liferay/portlet/admin/dependencies/email_
user_added_subject.tmpl
admin.email.user.added.body=com/liferay/portlet/admin/dependencies/email_use
r_added_body.tmpl
admin.email.password.sent.enabled=true
admin.email.password.sent.subject=com/liferay/portlet/admin/dependencies/ema
il_password_sent_subject.tmpl
admin.email.password.sent.body=com/liferay/portlet/admin/dependencies/email_
password_sent_body.tmpl
```

Announcements Portlet

Configure email notification settings.

```
announcements.email.from.name=Joe Bloggs
announcements.email.from.address=test@liferay.com
announcements.email.to.name=
announcements.email.to.address=noreply@liferay.com
announcements.email.subject=com/liferay/portlet/announcements/dependencies/e
mail_subject.tmpl
announcements.email.body=com/liferay/portlet/announcements/dependencies/emai
l_body.tmpl
```

Set the list of announcement types. The display text of each of the announcement types is set in `content/Language.properties`.

```
announcements.entry.types=general,news,test
```

Set the interval on which the CheckEntryJob will run. The value is set in one minute increments.

```
announcements.entry.check.interval=15
```

Blogs Portlet

The following properties affect the Blogs portlet.

```
blogs.email.comments.added.enabled=true
blogs.email.comments.added.subject=com/liferay/portlet/blogs/dependencies/em
ail_comments_added_subject.tmpl
blogs.email.comments.added.body=com/liferay/portlet/blogs/dependencies/email
_comments_added_body.tmpl
blogs.page.abstract.length=400
blogs.rss.abstract.length=200
blogs.trackback.excerpt.length=50
```

Set the interval on which the *TrackbackVerifierJob* will run. The value is set in one minute increments.

```
blogs.trackback.verifier.job.interval=5
```

Calendar Portlet

Set the list of event types. The display text of each of the event types is set in `content/Language.properties`.

```
calendar.event.types=anniversary,appointment,bill-payment,birthday,breakfast,
call,chat,class,club-event,concert,dinner,event,graduation,happyhour,
holiday,interview,lunch,meeting,movie,net-event,other,party,performance,
press-release,reunion,sports-event,training,travel,tvshow,
vacation,wedding
```

Set the interval on which the *CheckEventJob* will run. The value is set in one minute increments.

```
calendar.event.check.interval=15
```

Configure email notification settings.

```
calendar.email.from.name=Joe Bloggs
calendar.email.from.address=test@liferay.com
calendar.email.event.reminder.enabled=true
calendar.email.event.reminder.subject=com/liferay/portlet/calendar/dependencies/
email_event_reminder_subject.tmpl
calendar.email.event.reminder.body=com/liferay/portlet/calendar/dependencies/
email_event_reminder_body.tmpl
```

Communities Portlet

Configure email notification settings.

```
communities.email.from.name=Joe Bloggs
communities.email.from.address=test@liferay.com
communities.email.membership.reply.subject=com/liferay/portlet/communities/d
ependencies/email_membership_reply_subject.tmpl
communities.email.membership.reply.body=com/liferay/portlet/communities/dependencies/
email_membership_reply_body.tmpl
communities.email.membership.request.subject=com/liferay/portlet/communities
/dependencies/email_membership_request_subject.tmpl
communities.email.membership.request.body=com/liferay/portlet/communities/de
pendencies/email_membership_request_body.tmpl
```

Document Library Portlet

Set the name of a class that implements `com.liferay.documentlibrary.util.Hook`. The document library server will use this to persist documents.

Available hooks are:

- `com.liferay.documentlibrary.util.FileSystemHook`
- `com.liferay.documentlibrary.util.JCRHook`
- `com.liferay.documentlibrary.util.S3Hook`

Examples:

```
dl.hook.impl=com.liferay.documentlibrary.util.FileSystemHook
dl.hook.impl=com.liferay.documentlibrary.util.JCRHook
dl.hook.impl=com.liferay.documentlibrary.util.S3Hook
```

FileSystemHook

```
dl.hook.file.system.root.dir=${resource.repositories.root}/document_library
```

S3Hook

```
dl.hook.s3.access.key=
dl.hook.s3.secret.key=
dl.hook.s3.bucket.name=
```

Set the maximum file size and valid file extensions for documents. A value of 0 for the maximum file size can be used to indicate unlimited file size. However, the maximum file size allowed is set in the property `com.liferay.portal.upload.UploadServletRequestImpl.max.size`.

Examples:

```
#dl.file.max.size=307200
#dl.file.max.size=1024000
dl.file.max.size=3072000
```

A file extension of * will permit all file extensions.

You can map a GIF for the extension by adding the image to the theme's image display and document library folder. The wildcard extension of * will be ignored. For example, the default image for the DOC extension would be found in: `/html/themes/_unstyled/images/document_library/doc.gif`.

Example File Extensions:

```
dl.file.extensions=.bmp,.css,.doc,.dot,.gif,.gz,.htm,.html,.jpg,.js,.lar,.od
b,.odf,.odg,.odp,.ods,.odt,.pdf,.png,.ppt,.rtf,.swf,.sxc,.sxi,.sxw,.tar,.tif
f,.tgz,.txt,.vsd,.xls,.xml,.zip
```

Set which files extensions are comparable by the diff tool. If OpenOffice integration is enabled, then it is also possible to compare some binary files that are can be converted to text.

```
dl.comparable.file.extensions=.css,.js,.htm,.html,.txt,.xml
#dl.comparable.file.extensions=.css,.doc,.js,.htm,.html,.odt,.rtf,.sxw,.txt,
.xml
```

Set folder names that will be used to synchronize with a community's set of private and public layouts. This will allow users to manage layouts using the Document Library portlet, and ultimately, via WebDAV. This feature is experimental.

```
dl.layouts.sync.enabled=false
dl.layouts.sync.private.folder=Pages - Private
dl.layouts.sync.public.folder=Pages - Public
```

Image Gallery Portlet

Set the maximum file size and valid file extensions for images. A value of 0 for the maximum file size can be used to indicate unlimited file size. However, the maximum file size allowed is set in the property `com.liferay.portal.upload.UploadServletRequestImpl.max.size`.

```
ig.image.max.size=10240000
```

A file extension of * will permit all file extensions.

```
ig.image.extensions=.bmp,.gif,.jpeg,.jpg,.png,.tif,.tiff
```

Set the maximum thumbnail height and width in pixels. Set dimension of the custom images to 0 to disable creating a scaled image of that size.

```
ig.image.thumbnail.max.dimension=150
#ig.image.custom1.max.dimension=100
#ig.image.custom2.max.dimension=0
```

Invitation Portlet

```
invitation.email.max.recipients=20
invitation.email.message.body=com/liferay/portlet/invitation/dependencies/em
ail_message_body.tmpl
invitation.email.message.subject=com/liferay/portlet/invitation/dependencies/
email_message_subject.tmpl
```

Journal Portlet

Set this to true if article ids should always be autogenerated.

```
journal.article.force.autogenerate.id=true
```

Set this to true so that only the latest version of an article that is also not approved can be saved without incrementing version.

```
journal.article.force.increment.version=false
```

Set the list of article types. The display text of each of the article types is set in `content/Language.properties`.

```
journal.article.types=announcements,blogs,general,news,press-release,test
```

Set the token used when inserting simple page breaks in articles.

```
journal.article.token.page.break=@page_break@
```

Set the interval on which the *CheckArticleJob* will run. The value is set in one minute increments.

```
journal.article.check.interval=15
```

Set this to true if feed ids should always be autogenerated.

```
journal.feed.force.autogenerate.id=false
```

Set this to true if structure ids should always be autogenerated.

```
journal.structure.force.autogenerate.id=false
```

Set this to true if template ids should always be autogenerated.

```
journal.template.force.autogenerate.id=false
```

Input a comma delimited list of variables which are restricted from the context in Velocity based Journal templates.

```
journal.template.velocity.restricted.variables=serviceLocator
```

Set the maximum file size and valid file extensions for images. A value of 0 for the maximum file size can be used to indicate unlimited file size. However, the maximum file size allowed is set in the property `com.liferay.portal.upload.UploadServletRequestImpl.max.size`.

```
journal.image.small.max.size=51200
```

A file extension of * will permit all file extensions.

```
journal.image.extensions=.gif,.jpeg,.jpg,.png
```

Input a list of comma delimited class names that extend `com.liferay.portlet.journal.util.TransformerListener`. These classes will run in sequence to allow you to modify the XML and XSL before it's transformed and allow you to modify the final output.

```
journal.transformer.listener=\
com.liferay.portlet.journal.util.TokensTransformerListener,\
#com.liferay.portlet.journal.util.PropertiesTransformerListener,\
com.liferay.portlet.journal.util.ContentTransformerListener,\
com.liferay.portlet.journal.util.LocaleTransformerListener,\
com.liferay.portlet.journal.util.RegexTransformerListener,\
com.liferay.portlet.journal.util.ViewCounterTransformerListener
```

Enter a list of regular expression patterns and replacements that will be applied to outputted Journal content. The list of properties must end with a subsequent integer (0, 1, etc.) and it is assumed that the list has reached an end when the pattern or replacement is not set. See `com.liferay.portlet.journal.util.RegexTransformerListener` for implementation details.

```
#journal.transformer.regex.pattern.0=beta.sample.com
#journal.transformer.regex.replacement.0=production.sample.com
#journal.transformer.regex.pattern.1=staging.sample.com
#journal.transformer.regex.replacement.1=production.sample.com
```

Set whether to synchronize content searches when server starts.

```
journal.sync.content.search.on.startup=false
```

Configure mail notification settings.

```
journal.email.from.name=Joe Bloggs
journal.email.from.address=test@liferay.com
journal.email.article.approval.denied.enabled=false
journal.email.article.approval.denied.subject=com/liferay/portlet/journal/de
pendencies/email_article_approval_denied_subject.tmpl
journal.email.article.approval.denied.body=com/liferay/portlet/journal/dependencies/
email_article_approval_denied_body.tmpl
journal.email.article.approval.granted.enabled=false
journal.email.article.approval.granted.subject=com/liferay/portlet/journal/d
ependencies/email_article_approval_granted_subject.tmpl
journal.email.article.approval.granted.body=com/liferay/portlet/journal/dependencies/
email_article_approval_granted_body.tmpl
journal.email.article.approval.requested.enabled=false
journal.email.article.approval.requested.subject=com/liferay/portlet/journal
/dependencies/email_article_approval_requested_subject.tmpl
```

```
journal.email.article.approval.requested.body=com/liferay/portlet/journal/de
pendencies/email_article_approval_requested_body.tmpl
journal.email.article.review.enabled=false
journal.email.article.review.subject=com/liferay/portlet/journal/dependencies/
email_article_review_subject.tmpl
journal.email.article.review.body=com/liferay/portlet/journal/dependencies/e
mail_article_review_body.tmpl
```

Specify the strategy used when Journal content is imported using the LAR system.

```
journal.lar.creation.strategy=com.liferay.portlet.journal.lar.JournalCreationStrategyImpl
```

Specify the path to the template used for providing error messages on Journal templates.

```
journal.error.template.velocity=com/liferay/portlet/journal/dependencies/error.
vm
journal.error.template.xsl=com/liferay/portlet/journal/dependencies/error.
xsl
```

Journal Articles Portlet

Set the available values for the number of articles to display per page.

```
journal.articles.page.delta.values=5,10,25,50,100
```

Mail Portlet

Set the following to false if administrator should not be allowed to change the mail domain via the Admin portlet.

```
mail.mx.update=true
```

Set the name of a class that implements `com.liferay.mail.util.Hook`. The mail server will use this class to ensure that the mail and portal servers are synchronized on user information. The portal will not know how to add, update, or delete users from the mail server except through this hook.

Available hooks are:

- `com.liferay.mail.util.CyrusHook`
- `com.liferay.mail.util.DummyHook`
- `com.liferay.mail.util.FuseMailHook`
- `com.liferay.mail.util.SendmailHook`
- `com.liferay.mail.util.ShellHook`

Example:

```
mail.hook.impl=com.liferay.mail.util.DummyHook
```

CyrusHook

Set the commands for adding, updating, and deleting a user where %1% is the user id. Replace the password with the password for the cyrus user.

Add Examples:

```
mail.hook.cyrus.add.user=cyrusadmin password create %1%
mail.hook.cyrus.add.user=cyrus_adduser password %1%
```

Delete Examples:

```
mail.hook.cyrus.delete.user=cyrusadmin password delete %1%
mail.hook.cyrus.delete.user=cyrus_userdel password %1%
```

Other properties:

```
mail.hook.cyrus.home=/home/cyrus
```

FuseMailHook

See http://www.fusemail.com/support/api.html for more information. You must also update the `mail.account.finder` property.

```
mail.hook.fusemail.url=https://www.fusemail.com/api/request.html
mail.hook.fusemail.username=
mail.hook.fusemail.password=
mail.hook.fusemail.account.type=group_subaccount
mail.hook.fusemail.group.parent=
```

SendmailHook

Set the commands for adding, updating, and deleting a user where %1% is the user id and %2% is the password. Set the home and virtual user table information.

```
mail.hook.sendmail.add.user=adduser %1% -s /bin/false
mail.hook.sendmail.change.password=autopasswd %1% %2%
mail.hook.sendmail.delete.user=userdel -r %1%
mail.hook.sendmail.home=/home
mail.hook.sendmail.virtusertable=/etc/mail/virtusertable
mail.hook.sendmail.virtusertable.refresh=bash -c "makemap hash
/etc/mail/virtusertable < /etc/mail/virtusertable"
```

ShellHook

Set the location of the shell script that will interface with any mail server.

```
mail.hook.shell.script=/usr/sbin/mailadmin.ksh
```

Set to true to enable SMTP debugging.

```
mail.smtp.debug=false
```

Input a list of comma delimited email addresses that will receive a BCC of every email sent through the mail server.

```
mail.audit.trail=
```

Message Boards Portlet

Configure mail notification settings.

```
message.boards.email.from.name=Joe Bloggs
message.boards.email.from.address=test@liferay.com
message.boards.email.html.format=true
message.boards.email.message.added.enabled=true
message.boards.email.message.added.subject.prefix=com/liferay/portlet/messageboards/
dependencies/email_message_added_subject_prefix.tmpl
message.boards.email.message.added.body=com/liferay/portlet/messageboards/de
pendencies/email_message_added_body.tmpl
message.boards.email.message.added.signature=com/liferay/portlet/messageboards/
dependencies/email_message_added_signature.tmpl
message.boards.email.message.updated.enabled=true
message.boards.email.message.updated.subject.prefix=com/liferay/portlet/messageboards/
dependencies/email_message_updated_subject_prefix.tmpl
message.boards.email.message.updated.body=com/liferay/portlet/messageboards/
dependencies/email_message_updated_body.tmpl
message.boards.email.message.updated.signature=com/liferay/portlet/messageboards/
dependencies/email_message_updated_signature.tmpl
```

Set this to true to allow anonymous posting.

```
message.boards.anonymous.posting.enabled=true
```

Enter time in minutes on how often this job is run. If a user's ban is set to expire at 12:05 PM and the job runs at 2 PM, the expire will occur during the 2 PM run.

```
message.boards.expire.ban.job.interval=120
```

Enter time in days to automatically expire bans on users. Set to 0 to disable auto expire.

Examples:

```
message.boards.expire.ban.interval=10
message.boards.expire.ban.interval=0
```

Enter rss feed abstract length. This value limits what goes in the RSS feed from the beginning of the message board post. The default is the first 200 characters.

```
message.boards.rss.abstract.length=200
```

My Places Portlet

Set this to true to show user public sites with no layouts.

```
my.places.show.user.public.sites.with.no.layouts=true
```

Set this to true to show user private sites with no layouts.

```
my.places.show.user.private.sites.with.no.layouts=true
```

Set this to true to show organization public sites with no layouts.

```
my.places.show.organization.public.sites.with.no.layouts=true
```

Set this to true to show organization private sites with no layouts.

```
my.places.show.organization.private.sites.with.no.layouts=true
```

Set this to true to show community public sites with no layouts.

```
my.places.show.community.public.sites.with.no.layouts=true
```

Set this to true to show community private sites with no layouts.

```
my.places.show.community.private.sites.with.no.layouts=true
```

Navigation Portlet

Specify the options that will be provided to the user in the edit configuration mode of the portlet.

```
navigation.display.style.options=1,2,3,4,5,6
```

Define each mode with 4 comma delimited strings that represent the form: header- Type, rootLayoutType, rootLayoutLevel, and includedLayouts.

```
navigation.display.style[1]=breadcrumb,relative,0,auto
navigation.display.style[2]=root-layout,absolute,2,auto
navigation.display.style[3]=root-layout,absolute,1,auto
navigation.display.style[4]=none,absolute,1,auto
navigation.display.style[5]=none,absolute,1,all
navigation.display.style[6]=none,absolute,0,auto
```

Nested Portlets Portlet

```
nested.portlets.layout.template.default=2_columns_i
```

Add a comma separated list of layout template ids that should not be allowed in the Nested Portlets Portlet.

```
nested.portlets.layout.template.unsupported=freeform,1_column
```

Portlet CSS Portlet

Set this to true to enable the ability to modify portlet CSS at runtime via the Look and Feel icon. Disabling it can speed up performance.

```
portlet.css.enabled=true
```

Shopping Portlet

Set the following to true if cart quantities must be a multiple of the item's minimum quantity.

```
shopping.cart.min.qty.multiple=true
```

Set the following to true to forward to the cart page when adding an item from the category page. The item must not have dynamic fields. All items with dynamic fields will forward to the item's details page regardless of the following setting.

```
shopping.category.forward.to.cart=false
```

Set the following to true to show special items when browsing a category.

```
shopping.category.show.special.items=false
```

Set the following to true to show availability when viewing an item.

```
shopping.item.show.availability=true
```

Set the maximum file size and valid file extensions for images. A value of 0 for the maximum file size can be used to indicate unlimited file size. However, the maximum file size allowed is set in the property `com.liferay.portal.upload.UploadServletRequestImpl.max.size`.

```
shopping.image.small.max.size=51200
shopping.image.medium.max.size=153600
shopping.image.large.max.size=307200
```

A file extension of * will permit all file extensions.

```
shopping.image.extensions=.gif,.jpeg,.jpg,.png
```

Configure email notification settings.

```
shopping.email.from.name=Joe Bloggs
shopping.email.from.address=test@liferay.com
shopping.email.order.confirmation.enabled=true
shopping.email.order.confirmation.subject=com/liferay/portlet/shopping/dependencies/
email_order_confirmation_subject.tmpl
shopping.email.order.confirmation.body=com/liferay/portlet/shopping/dependencies/
email_order_confirmation_body.tmpl
shopping.email.order.shipping.enabled=true
shopping.email.order.shipping.subject=com/liferay/portlet/shopping/dependencies/
email_order_shipping_subject.tmpl
shopping.email.order.shipping.body=com/liferay/portlet/shopping/dependencies/
email_order_shipping_body.tmpl
```

Software Catalog Portlet

Set the maximum file size and max file dimensions for thumbnnails. A value of 0 for the maximum file size can be used to indicate unlimited file size. However, the maximum file size allowed is set in the property `com.liferay.portal.upload.UploadServletRequestImpl.max.size`.

```
sc.image.max.size=307200
sc.image.thumbnail.max.height=200
sc.image.thumbnail.max.width=160
```

Tags Compiler Portlet

Set this to true to enable the ability to compile tags from the URL. Disabling it can speed up performance.

```
tags.compiler.enabled=true
```

Tags Portlet

Input a class name that implements `com.liferay.portlet.tags.util.TagsAssetValidator`. This class will be called to validate assets. The `DefaultTagsAssetValidator` class is just an empty class that doesn't actually do any validation.

The `MinimalTagsAssetValidator` requires all assets to have at least one tag entry.

Examples:

```
tags.asset.validator=com.liferay.portlet.tags.util.DefaultTagsAssetValidator
#tags.asset.validator=com.liferay.portlet.tags.util.MinimalTagsAssetValidator
```

Tasks Portlet

Specify the default number of approval stages.

```
tasks.default.stages=2
```

Specify the default role name for each stage of approval ordered from lowest level of approval to highest. These Roles must have the APPROVE_PROPOSAL permission.

```
tasks.default.role.names=Community Administrator,Community Owner
```

Translator Portlet

Set the default languages to translate a given text.

```
translator.default.languages=en_es
```

Web Form Portlet

Set the maximum number of dynamic fields to process.

```
web.form.portlet.max.fields=50
```

Wiki Portlet

Set the URL of a page that contains more information about the classic syntax of the wiki. It will be shown to the user when editing a page.

```
wiki.classic.syntax.help.url=http://wiki.liferay.com/index.php/Wiki_Portlet
```

Set the name of the default page for a wiki node. The name for the default page must be a valid wiki word. A wiki word follows the format of having an upper case letter followed by a series of lower case letters followed by another upper case letter and another series of lower case letters. See http://www.use mod.com/cgi-bin/wiki.pl?WhatIsaWiki for more information on wiki naming conventions.

```
wiki.front.page.name=FrontPage
```

Set the name of the default node that will be automatically created when the Wiki portlet is first used in a community.

```
wiki.initial.node.name=Main
```

Set the following property to specify the requirments for the names of wiki pages. By default only a few characters are forbidden. Uncomment the regular expression below to allow only *CamelCase* titles.

```
wiki.page.titles.regexp=([^/\\[\\]%&?@]+)
#wiki.page.titles.regexp=(((\\p{Lu}\\p{Ll}+)_?)+)
```

Set the following property to specify the characters that will be automatically removed from the titles when importing wiki pages. This *regexp* should remove any characters that are forbidden in the regexp specified in `wiki.page.titles.regexp`.

```
wiki.page.titles.remove.regexp=([/\\[\\]%&?@]+)
```

Set the list of supported wiki formats and the default wiki format.

```
wiki.formats=creole,html
wiki.formats.default=creole
```

Configure settings for each of the wiki formats.

```
wiki.formats.engine[classic_wiki]=com.liferay.portlet.wiki.engines.friki.
FrikiEngine
wiki.formats.configuration.main[classic_wiki]=wiki.transform
wiki.formats.configuration.interwiki[classic_wiki]=intermap.txt
wiki.formats.edit.page[classic_wiki]=/html/portlet/wiki/edit/wiki.jsp
wiki.formats.help.page[classic_wiki]=/html/portlet/wiki/help/classic_wiki.
jsp
wiki.formats.help.url[classic_wiki]=http://wiki.liferay.com/index.php/Wiki_P
ortlet

wiki.formats.engine[creole]=com.liferay.portlet.wiki.engines.jspwiki.JSPWikiEngine
wiki.formats.configuration.main[creole]=jspwiki.properties
wiki.formats.edit.page[creole]=/html/portlet/wiki/edit/wiki.jsp
wiki.formats.help.page[creole]=/html/portlet/wiki/help/creole.jsp
wiki.formats.help.url[creole]=http://www.wikicreole.org/wiki/Creole1.0
wiki.formats.engine[html]=com.liferay.portlet.wiki.engines.HtmlEngine
wiki.formats.edit.page[html]=/html/portlet/wiki/edit/html.jsp
wiki.formats.engine[plain_text]=com.liferay.portlet.wiki.engines.TextEngine
wiki.formats.edit.page[plain_text]=/html/portlet/wiki/edit/plain_text.jsp
```

Set the list of supported wiki importers.

```
wiki.importers=MediaWiki
```

Configure settings for each of the wiki importers.

```
wiki.importers.page[MediaWiki]=/html/portlet/wiki/import/mediawiki.jsp
wiki.importers.class[MediaWiki]=com.liferay.portlet.wiki.importers.mediawiki.
MediaWikiImporter
```

Configure email notification settings.

```
wiki.email.from.name=Joe Bloggs
wiki.email.from.address=test@liferay.com
wiki.email.page.added.enabled=true
wiki.email.page.added.subject.prefix=com/liferay/portlet/wiki/dependencies/e
mail_page_added_subject_prefix.tmpl
wiki.email.page.added.body=com/liferay/portlet/wiki/dependencies/email_page_
added_body.tmpl
wiki.email.page.added.signature=com/liferay/portlet/wiki/dependencies/email_
page_added_signature.tmpl
wiki.email.page.updated.enabled=true
wiki.email.page.updated.subject.prefix=com/liferay/portlet/wiki/dependencies/
email_page_updated_subject_prefix.tmpl
wiki.email.page.updated.body=com/liferay/portlet/wiki/dependencies/email_pag
e_updated_body.tmpl
wiki.email.page.updated.signature=com/liferay/portlet/wiki/dependencies/emai
l_page_updated_signature.tmpl
wiki.rss.abstract.length=200
```

WebSynergy Clustering

Once you have WebSynergy installed in more than one node on your application server, there are several optimizations that need to be made. At a minimum, WebSynergy should be configured in the following way for a clustered environment:

Note – The default HSQL database can't be used in a clustered environment. You configure MySQL or any other compatible database to use in a clustered environment.

- All nodes should be pointing to the same Liferay database

- Jackrabbit, the JSR-170 content repository, should be on a shared file system (not recommended) or in a database that is shared by all nodes.
- Similarly, Lucene, the full text search indexer, should be:
  - On a shared file system available to all the nodes (not really recommended, though), or
  - In a database that is shared by all the nodes, or
  - On separate file systems for all of the nodes, or
  - Disabled, and a separate pluggable enterprise search server configured.
- If you have not configured your application server to use farms for deployment, the hot deploy folder should be a separate folder for all the nodes, and plugins will have to be deployed to all of the nodes individually. This can be done via a script.

Many of these configuration changes can be made by adding or modifying properties in your `portal-ext.properties` file. Remember that this file overrides the defaults that are in the `portal.properties` file. The original version of this file can be found in the Liferay source code or can be extracted from the `portal-impl.jar` file in your Liferay installation. It is a best practice to copy the relevant section that you want to modify from `portal.properties` into your `portal-ext.properties` file, and then modify the values there.

- "Jackrabbit Sharing" on page 163
- "Lucene Configuration" on page 164
- "Hot Deploy" on page 165

Jackrabbit Sharing

WebSynergy uses Jackrabbit from Apache as its JSR-170 compliant document repository. By default, Jackrabbit is configured to store the documents on the local file system upon which Liferay is installed, in the <*Glassfish home*>`/domains/domain1/websynergy/jackrabbit` folder. Inside this folder is Jackrabbit's configuration file, called `repository.xml`.

To simply move the default repository location to a shared folder, you do not need to edit Jackrabbit's configuration file. Instead, find the section in `portal.properties` labeled *JCR* and copy/paste that section into your `portal-ext.properties` file. One of the properties, by default, is the following:

```
jcr.jackrabbit.repository.root=${resource.repositories.root}/jackrabbit
```

Change this property to point to a shared folder that all of the nodes can see. A new Jackrabbit configuration file will be generated in that location.

Note that because of file locking issues, this is not the best way to share Jackrabbit resources. If you have two people logged in at the same time uploading content, you could encounter data corruption using this method, and because of this, we do not recommend it for a production system. Instead, to enable better data protection, you should redirect Jackrabbit into your database of choice. You can use a database for this purpose. This will require editing Jackrabbit's configuration file.

The default Jackrabbit configuration file has sections commented out for moving the Jackrabbit configuration into the database. This has been done to make it as easy as possible to enable this configuration. To move the Jackrabbit configuration into the database, simply comment out the sections relating to the file system and comment in the sections relating to the database. These by default are configured for a MySQL database. If you are using another database, you will likely need to modify the configuration, as there are changes to the configuration file that are necessary for specific databases. For example, the default configuration uses Jackrabbit's `DbFileSystem` class to mimic a file system in the database. While this works well in MySQL, it does not work for all databases. For example, if you are using an Oracle database, you will need to modify this to use `OracleFileSystem`. Please see the Jackrabbit documentation at http://jackrabbit.apache.org for further information.

You will also likely need to modify the JDBC database URLs so that they point to your database. Don't forget to create the database first, and grant the user ID you are specifying in the configuration file access to create, modify, and drop tables.

Once you have configured Jackrabbit to store its repository in a database, the next time you bring up Liferay, the necessary database tables will be created automatically. Jackrabbit, however, does not create indexes on these tables, and so over time this can be a performance penalty. To fix this, you will need to manually go into your database and index the primary key columns for all of the Jackrabbit tables.

All of your Liferay nodes should be configured to use the same Jackrabbit repository in the database. Once that is working, you can create a Jackrabbit cluster (please see the section below).

Lucene Configuration

Lucene, the search indexer which WebSynergy uses, can be in a shared configuration for a clustered environment, or an index can be created on each node of the cluster. If you wish to have a shared index, you will need to either share the index on the file system or in the database.

The Lucene configuration can be changed by modifying values in your `portal-ext.properties` file. Open your `portal.properties` file and search for the text Lucene. Copy that section and then paste it into your `portal-ext.properties` file.

If you wish to store the Lucene search index on a file system that is shared by all of the WebSynergy nodes, you can modify the location of the search index by changing the `lucene.dir` property. By default, this property points to the `/websynergy/lucene` folder inside the home folder of the user running WebSynergy:

```
lucene.dir=${resource.repositories.root}/lucene/
```

Change this to the folder of your choice. To make the change take effect, you will need to restart WebSynergy. You can point all of the nodes to this folder, and they will use the same index.

Like Jackrabbit, however, this is not the best way to share the search index, as it could result in file corruption if different nodes try reindexing at the same time. We do not recommend this for a production system. A better way is to share the index via a database, where the database can enforce data integrity on the index. This is very easy to do; it is a simple change to your `portal-ext.properties` file.

There is a single property called `lucene.store.type`. By default this is set to go to the file system. You can change this so that the index is stored in the database by making it the following:

```
lucene.store.type=jdbc
```

The next time WebSynergy is started, new tables will be created in the WebSynergy database, and the index will be stored there. If all the WebSynergy nodes point to the same database tables, they will be able to share the index.

Alternatively, you leave the configuration alone, and each node will then have its own index. This ensures that there are no collisions when multiple nodes update the index, because they all will have separate indexes.

Hot Deploy

Plugins which are hot deployed will need to be deployed separately to all of the WebSynergy nodes. Each node should, therefore, have its own hot deploy folder. This folder needs to be writable by the user under which WebSynergy is running, because plugins are moved from this folder to a temporary folder when they are deployed. This is to prevent the system from entering an endless loop, because the presence of a plugin in the folder is what triggers the hot deploy process.

When you want to deploy a plugin, copy that plugin to the hot deploy folders of all of the WebSynergy nodes. The hot deploy directory for WebSynergy when running on GlassFish is <*Glassfish home*>`/domains/domain1/websynergy/deploy`. Depending on the number of nodes, it may be best to create a script to do this. Once the plugin has been deployed to all of the nodes, you can then make use of it (by adding the portlet to a page or choosing the theme as the look and feel for a page or page hierarchy).

Some containers contain a facility which allows the end user to deploy an application to one node, after which it will get copied to all of the other nodes. If you have configured your application server to support this, you won't need to hot deploy a plugin to all of the nodes, as your application server will handle it transparently. Make sure, however, that you use hot deploy mechanism to deploy plugins, as in many cases WebSynergy slightly modifies `plugin.war` files when hot deploying them.

Installing Plugins

WebSynergy comes with two portlets which can handle plugin installation: the *Plugin Installer* and the *Update Manager*. The Update Manager helps to determine if you are running the most recent version of a plugin.

You can add the Update Manager portlet to your page by clicking Add Application from the welcome dock. The Update Manager displays which plugins are already installed on the system, what their version numbers are, and whether an update is available.

To install a plugin from the *Update Manager*, click on the *Install More Plugins* button. It invokes the *Plugin Installer* portlet, and by default you are on the Portlet Plugins tab. You can install or uninstall the portlets available in the repository. If your server is firewalled, you may not see any plugins in the repository, and you need to install plugins manually. To install plugins manually, click on the *Upload File* tab. You can browse the WAR file for a layout template, portlet, or a theme that you want to install. You can specify the deployment context in a text box for easy identification of the portlet. Click on the *Install* button to install the portlet.

If you do not wish to use the Update Manager or Plugin Installer to deploy plugins, you can also deploy them at the operating system level. The first time WebSynergy starts, it creates a hot deploy folder which is by default created inside the home folder of the user who launched WebSynergy. For example, say that on a Linux system, the user `lportal` was created in order to run WebSynergy. The first time WebSynergy is launched, it will create a folder structure in `/home/lportal/websynergy` to house various configuration and administrative data. One of the folders it creates is called `deploy`. If you copy a portlet or theme plugin into this folder, Liferay will deploy it and make it available for use just as though you'd installed it via the Update Manager or Plugin Installer. In fact, this is what the Update Manager and Plugin Installer portlets are doing behind the scenes.

You can change the defaults for this directory structure so that it is stored anywhere you like by modifying the appropriate properties in your `portal-ext.properties` file.

Creating a Custom Plugin Repository

As your enterprise builds its own library of portlets for internal use, you can create your own plugin repository to make it easy to install and upgrade portlets. This will allow different departments running different instances of WebSynergy to share portlets and install them as needed. If you are a software development house, you may wish to create a plugin repository for your own products. WebSynergy makes it easy for you to create your own plugin repository and make it available to others.

You can create your plugin repository using the *Software Catalog* portlet. This method allows users to upload their plugins to an HTTP server to which they have access. They can then register their plugins with the repository by adding a link to it via the portlet's graphical user interface. WebSynergy will then generate the XML necessary to connect the repository to a *Plugin Installer* portlet running another instance of WebSynergy. This XML file can then be placed on an HTTP server, and the URL to it can be added to the *Plugin Installer*, making the portlets in this repository available to the server running WebSynergy.

Using the Software Catalog Portlet

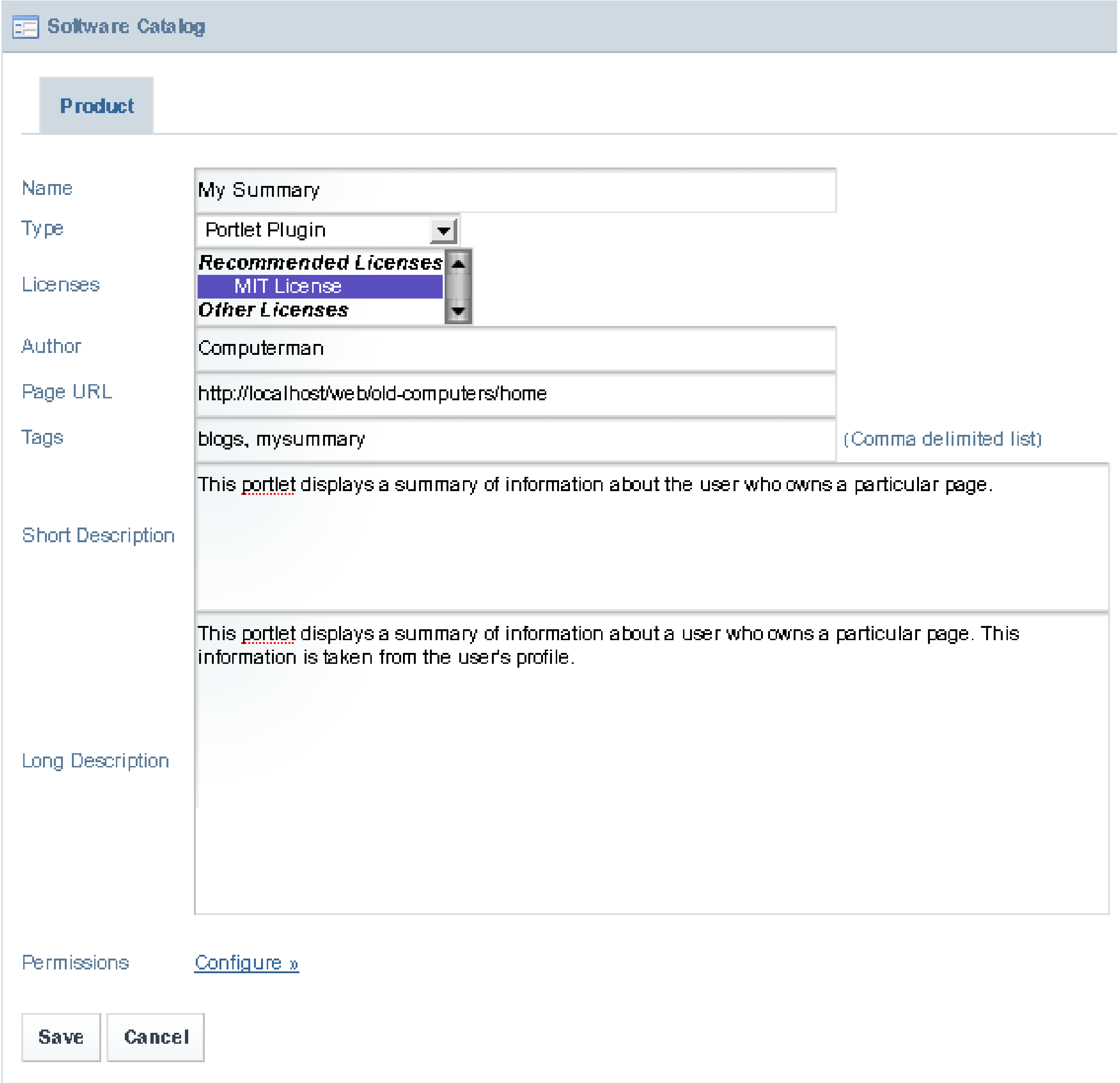
The *Software Catalog* portlet is not an instanceable portlet, which means that each community can have only one instance of the portlet. If you add the portlet to another page in the community, it will hold the same data as the portlet that was first added. Different communities, however, can have different software repositories, so you can host several software repositories on the same instance of WebSynergy if you wish they just have to be in different communities.

The *Software Catalog* portlet has several tabs. The first tab is labeled *Products*. The default view of the portlet, when populated with software, displays what plugins are available for install or download. This can be seen in the version on WebSynergy's home page.

The first step in adding a plugin to your software repository is to add a license for your product. A license communicates to users the terms upon which you are allowing them to download and use your software. Click the *Licenses* tab and then click the *Add License* button that appears. You will then see a form which allows you to enter the title of your license, a URL pointing to the actual license document, and check boxes denoting whether the license is open source, active, or recommended.

When you have finished filling out the form, click the *Save* button. Your license will be saved. Once you have at least one license in the system, you can begin adding software products to your software catalog. Your next step will be to create the product record in the software catalog portlet. This will register the product in the software catalog and allow you to start adding versions of your software for users to download and/or install directly from their instances of WebSynergy. You will first need to put the `.war` file containing your software on a web server that is accessible without authentication to the users who will be installing your software. If you are creating a software catalog for an internal Intranet, you would place the file on a web server that is available to anyone inside of your organization's firewall.

To create the product record in the Software Catalog portlet, click the *Products* tab, and then click the *Add Product* button. Fill out the form with information about your product.



Software Catalog

Product

Name: My Summary

Type: Portlet Plugin

Licenses:
Recommended Licenses
MIT License
Other Licenses

Author: Computerman

Page URL: http://localhost/web/old-computers/home

Tags: blogs, mysummary (Comma delimited list)

Short Description: This portlet displays a summary of information about the user who owns a particular page.

Long Description: This portlet displays a summary of information about a user who owns a particular page. This information is taken from the user's profile.

Permissions: Configure »

FIGURE 5–1 Adding a Product to the Software Catalog (partial)

Name: The name of your software product.

Type: Select whether this is a portlet or a theme plugin.

Licenses: Select the license(s) under which you are releasing this software.

Author: Enter the name of the author of the software.

Page URL: If the software has a home page, enter its url here.

Tags: Enter any tags you would like added to this software.

Short Description: Enter a short description. This will be displayed in the summary table of your software catalog.

Long Description: Enter a longer description. This will be displayed on the details page for this software product.

Permissions: Click the Configure link to set permissions for this software product.

Group ID: Enter a group ID. A group ID is a name space which usually identifies the company or organization that made the software. For our example, we will use old-computers.

Artifact ID: Enter an Artifact ID. The artifact ID is a unique name within the name space for your product. For our example, we will use my-summary-portlet.

Screenshot: Click the Add Screenshot button to add a screenshot of your product for users to view.

When you have finished filling out the form, click the *Save* button. You will be brought back to the product summary page, and you will see that your product has been added to the repository.

Notice that in the version column, N/A is being displayed. This is because there are not yet any released versions of your product. To make your product downloadable, you need to create a version of your product and point it to the file you uploaded to your HTTP server earlier.

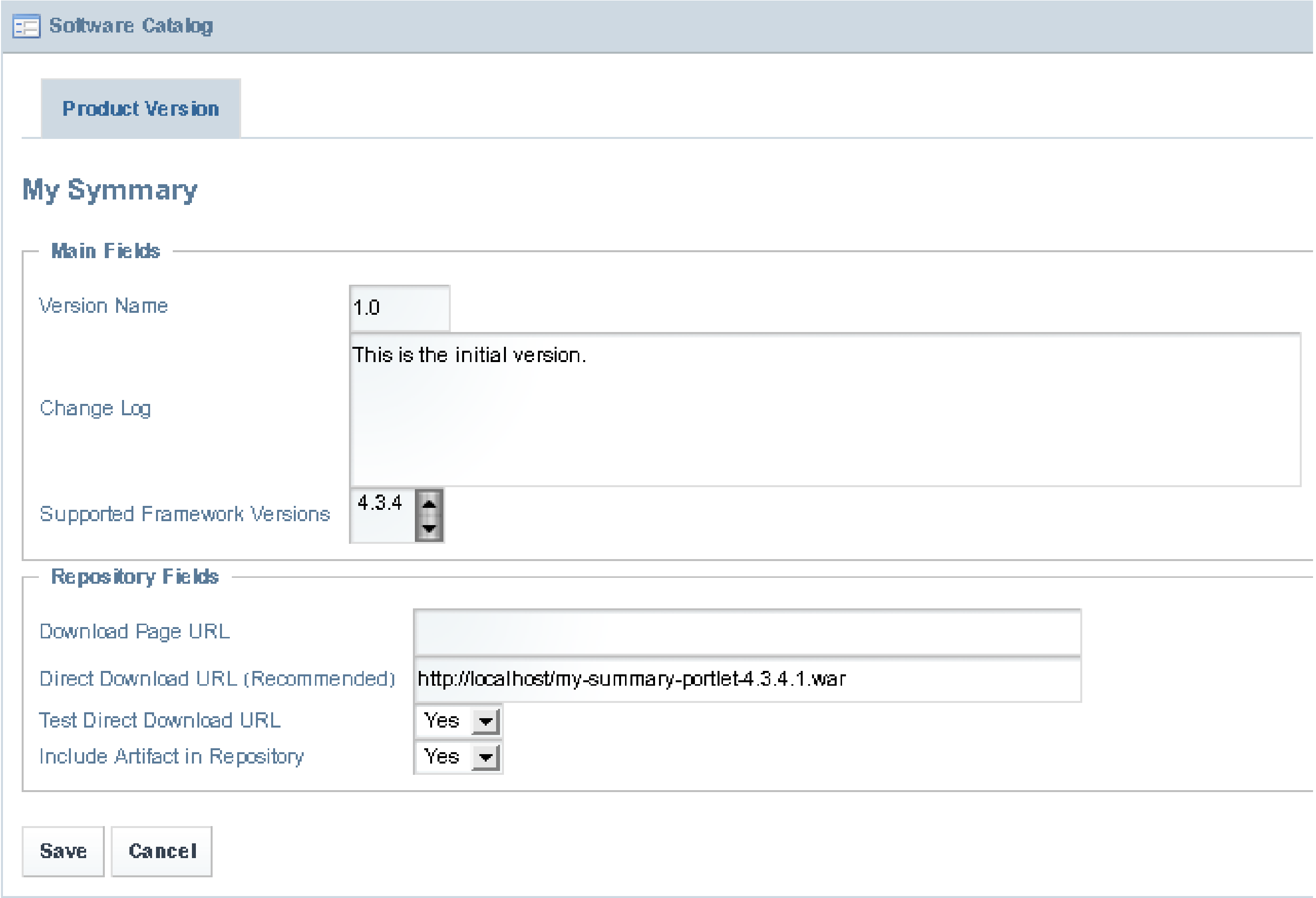
Before you do that, however, you need to add a *Framework Version* to your software catalog. A Framework version denotes what version of WebSynergy your plugin is designed for and works on. You cannot add a version of your product without linking it to a version of the framework for which it is designed.

Why is this so important? Because as WebSynergy gains more and more features, you may wish to take advantage of those features in future versions of your product, while still keeping older versions of your product available for those who are using older versions of WebSynergy.

So click the *Framework Versions* tab and then click the *Add Framework Version* button. Give the framework a name, a URL, and leave the Active check box checked.

Now go back to the *Products* tab and click on your product. You will notice that a message is displayed stating that the product does not have any released versions. Click the *Add Product Version* button.

Note – It is a must to specify a group ID and artifact ID before you specify a product version. You can specify the group ID and artifact ID for the product from the Product Version page by clicking on the *You must to specify a group ID and artifact ID before you specify a product version* link, which appears in the Product Version page if a group ID and artifact ID are not specified for the product.



Version Name	1.0
Change Log	This is the initial version.
Supported Framework Versions	4.3.4
Download Page URL	
Direct Download URL (Recommended)	http://localhost/my-summary-portlet-4.3.4.1.war
Test Direct Download URL	Yes
Include Artifact in Repository	Yes

FIGURE 5–2 Adding a product version to the software catalog

Version Name: Enter the version of your product.

Change Log: Enter some comments regarding what changed between this version and any previous versions.

Supported Framework Versions: Select the framework version for which your software product is intended.

Download Page URL: If your product has a descriptive web page, enter its URL here.

Direct Download URL (Recommended): Enter a direct download link to your software product here. The Plugin Installer portlet will follow this link in order to download your software product.

Include Artifact in Repository: To enable others to use the Plugin Installer portlet to connect to your repository and download your plugin, select *Yes* here.

When you are finished filling out the form, click the *Save* button. Your product version will be saved, and your product will now be available in the software repository.

CHAPTER 6

Administering Content Management System

The CMS (Content Management System) feature makes it possible to create, edit, and publish documents and articles online. It makes it possible to maintain online repositories of documentation for storing, controlling, versioning, and publishing of information.

- Content Management in WebSynergy
- Workflow Portlet

Content Management in WebSynergy

Liferay Portal provides out-of-the-box content management (CMS) functionality in two major areas: document management (repositories for documents and images) and publishing.

Document management is provided through WebSynergy's Document Library and Image Gallery portlets. These portlets provide intuitive management of Microsoft Office documents, PDFs, text files, and images.

Publishing in the Project WebSynergy portal is accomplished through the Web Content portlet. It features a clean separation between content (data entry) and presentation (layout design). This allows the content to conform to a consistent "look and feel." Additionally, the Web Content portlet provides a content versioning and approval paradigm (workflow). This system allows for users involved in publishing to fall naturally into three different roles:

1. Writers create, edit, and enter content comprised of text and images.
2. Designers create layouts that display content.
3. Editors edit and approve final output and release it for publication to the portal.

As an example, consider the Project WebSynergy website. All of the static content is managed and published by the *Web Content* portlet. We can authorize other members of the WebSynergy community to write documentation, but we retain final editorial authority and security. This also allows for a mixture of portlets and Journal articles to coexist, enabling the designer to be more creative with what is displayed. Designers can integrate static content using the *Journal Content* portlet with a number of other function-rich portlets such as *Navigation* and *Breadcrumb*. All of this shows how useful and flexible publishing can be with the *Web Content* portlet.

Document Management portlets:

- "Document Library Portlet" on page 174
- "Document Library Display Portlet" on page 176
- "Image Gallery Portlet" on page 176

Web Content portlet:

- "Web Content portlets" on page 180
- "Web Content List Portlet" on page 191
- "Web Content Display Portlet" on page 191
- "Web Content Search Portlet" on page 193

Other portlets:

- "Asset Publisher Portlet" on page 195
- "Breadcrumb Portlet" on page 211
- "Categories Navigation Portlet" on page 211
- "Navigation Portlet" on page 212
- "Nested Portlets Portlet" on page 212
- "Recent Documents Portlet" on page 212
- "Site Map Portlet" on page 212
- "Tags Admin Portlet" on page 212
- "XSL Content Portlet" on page 215

Document Library Portlet

The Document Library provides a centralized repository for various document types such as .PDF, .DOC, and .XLS, to be stored and assigns a unique URL to each of them. A company's intranet running Project WebSynergy portal can then be able to post internal documentation such as reimbursement forms and charitable donations policies for access by its employees. Alternatively, since each document has its own URL, the *Asset Publisher* portlet on your company's public website may display a list of PDF files of quarterly newsletters and their download links.

The Document Library supports multiple repositories that can be set with different access roles. Also, the documents in the repository are version-controlled, allowing a user to retrieve any particular revision that has been stored on the system.

- "To add a Document" on page 175
- "To create Reference to a Document" on page 175

▼ To add a Document

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**
4. **Click Add against the Document Library Portlet.**

   The Document Library Portlet is added to your page.

   The user interface of the portlet contains *Folders*, *My Documents*, and *Recent Documents* tabs. You can search documents using the *Search File Entries* button in the *Folders* tab. The *Add Folder* button in the *Folders* tab allows you to add folders. *My Documents* and *Recent Document* tabs list all the documents and the most recently accessed documents respectively.
5. **Any document need to be in a folder. To add a new folder, click the Add Folder button.**
6. **Give a name and description to the folder. The permissions for the folder determine what users can do. Click Save.**

   A folder is created.
7. **To add a document to that folder, click on the folder name.**
8. **You can either add more folders to further divide your documents into more specific categories or you can add a document to the current folder. In this example, we add a document. Click Add Document.**
9. **Browse and upload a document using the classic uploader.**

   There is a cap of 3000k on uploaded documents. After uploading the document, enter a title and description for the document.
10. **Click Save when finished.**

    The document is added to the Document Library.

▼ To create Reference to a Document

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**

4. **Click Add against the Document Library Portlet.**

   The Document Library Portlet is added to your page.

5. **Locate a document in the Document Library.**

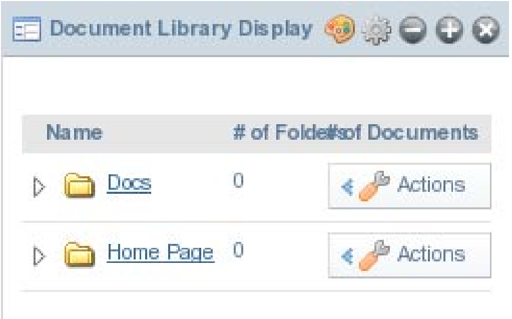
6. **Right click on the document name, then select Properties to view its Address (URL).**

   You can then either highlight the entire URL or click anywhere over the URL area and press Ctrl+A to select the entire link. Keep in mind that the URL is long. If you use the highlight method, you must highlight the entire URL. You can then copy this link and reference it anywhere within the portal. You can even reference this document outside of the portal as long as your portal is running and the document is still in the repository. Refer to figure below.

Document Library Display Portlet

The Document Library Display Portlet holds the display functionality for Document Library. You can navigate to the folders and document in the Document Library, even by accessing the Document Library Display Portlet.

You can add Document Library Display Portlet to your page from among Content Management portlets in the Add Applications window.



Document Library Display

Name	# of Folde#sof Documents	
Docs	0	Actions
Home Page	0	Actions

FIGURE 6–1 Document Library Display Portlet

Image Gallery Portlet

Like the Document Library, the Image Gallery provides a centralized repository for images used throughout the Project WebSynergyPortal and assigns a unique URL to each image.

- "To add an Image" on page 177
- "To create Reference to an Image" on page 177
- "To View Image Slide Show" on page 178

▼ To add an Image

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**
4. **Click Add against the Image Gallery Portlet.**

   The portlet is added to your page.
5. **Click the Add Folder button.**

   Specify a name and description and click **Save**. The folder is added to the portlet.
6. **Click the folder name.**

   You can add a subfolder or an Image within the folder. In our example we are adding an image for the folder we have already created.
7. **Click the Add Image button.**
8. **Use the classic uploader to upload the image.**

   There is a cap of 3000k on uploaded documents. After uploading the image, enter a title and description for the image.

▼ To create Reference to an Image

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**
4. **Click Add against the Image Gallery Portlet.**

   The portlet is added to your page.
5. **Locate an image in the Image Gallery.**
6. **Right click on the image, then select Properties to view its Address (URL).**

   You can then either highlight the entire URL or click anywhere over the URL area and press Ctrl+A to select the entire link. Keep in mind that the URL is long. If you use the highlight method, you must highlight the entire URL. You can then copy this link and reference it anywhere within the portal. You can even reference this document outside of the portal as long as your portal is running and the document is still in the repository.

▼ To View Image Slide Show

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**
4. **Click Add against the image gallery portlet.**

   The portlet is added to your page.
5. **Click on a folder containing images.**

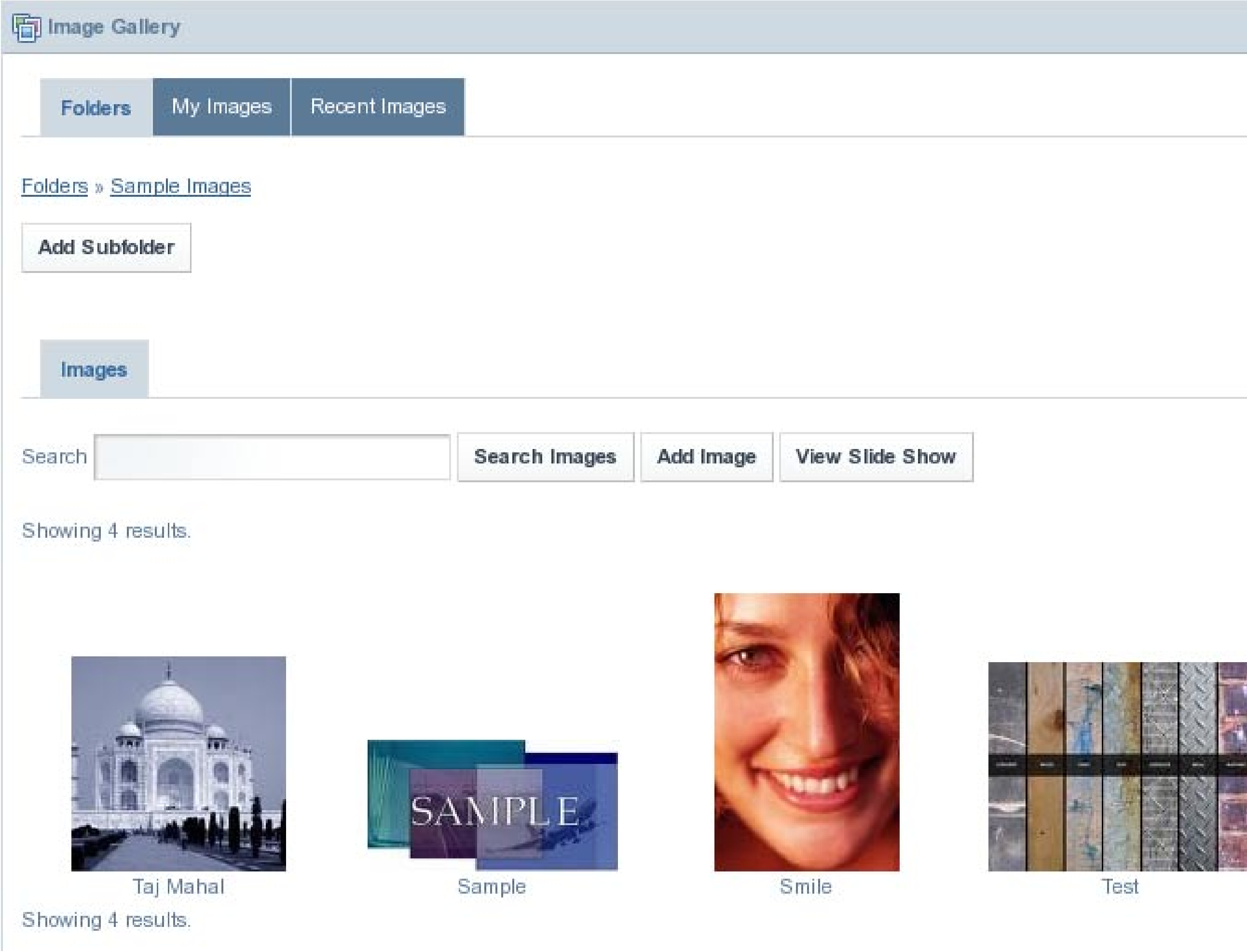


Image Gallery

Folders | My Images | Recent Images

Folders » Sample Images

Add Subfolder

Images

Search | Search Images | Add Image | View Slide Show

Showing 4 results.

Showing 4 results.

FIGURE 6–2 To view Image Slide Show

6. **Click on the View Slide Show button to have a slide show of the images contained in the folder.**

   A new window appears. In the new window, images are displayed in a sequence, one after the other. The slide show window has buttons for *Pause*, *Play*, *Next*, and *Previous*, which allow you to play and pause the slide show, and to view next and previous image for an image. You can set a time interval for displaying images, by using the Speed list. You can set a time interval between 1 to 10 seconds, with 3 seconds being the default time interval for the slide show.

Web Content portlets

The Web Content application is composed of four portlets:

Web Content Is an administration portlet that allows the creating, updating searching and managing contents.

Web Content Display Allows displaying of a set of contents that satisfy a set of conditions configured through the portlet preferences such as content type and community.

Web Content List Allows displaying one or more contents manually selected from the list of available portlets. If more than one content is selected only one at a time will be shown and links will be added to allow the user to navigate to the rest of the contents.

Web Content Search Allows users to search for web content.

the Web Content portlet is used to create all the CMS content and the Web Content Display and Web Content List portlets are used to display those contents. This portlet allows writers to easily edit content displayed on web pages without having to sort through complex code.

Definitions

Before getting into the details of how to use the Web Content portlet, it is useful to understand structures, templates, and articles; they are the backbone of the Web Content portlet's functionality.

- *Structures* define the type and number of content pieces (e.g. text, text area, images, selection lists, etc.)
- *Templates* determine how content pieces will be arranged or laid out
- *Articles* are where each content piece is populated with actual text and images

The point of a structure is to unify journal articles with the same numbers and types of items. For example, a Management Bio section of a website might have ten biographies, each requiring an image for the executive headshot, a text for the headline, and a text area for the main biography. A structure would be created with one image, one text, and one text area, each named accordingly. This allows writers to create the individual biographies and other texts without needing to recreate the page structure for each biography.

Designers can create several templates for one structure to allow for a variety of presentations or they can give writers discretion in deciding the best layout.

All specific text and images are entered when a user adds or edits an article. This is also where a user chooses the layout based on the templates affiliated with a structure. Since multiple templates can be associated to a structure, the user can select the template that formats their article the way they want.

▼ To add a Web Content portlet to the Page

1. **Login to WebSynergy.**

2. **Click Add Applications from the welcome menu.**

3. **Expand the Content Management folder.**

4. **Click Add against the Web Content portlet.**

   The Web Content portlet is added to your page.

   The user interface of the portlet contains *Articles*, *Structures*, *Templates*, *Feeds* and *Recent* tabs. You can search documents using the *Search Articles* button in the *Articles* tab. You can add an article using the Add Article button.

5. **Click the Articles tab to view all currently available Journal Articles.**

Structures

- "To create a Structure" on page 181
- "To Edit a Structure" on page 183
- "To Delete a Structure" on page 184

▼ To create a Structure

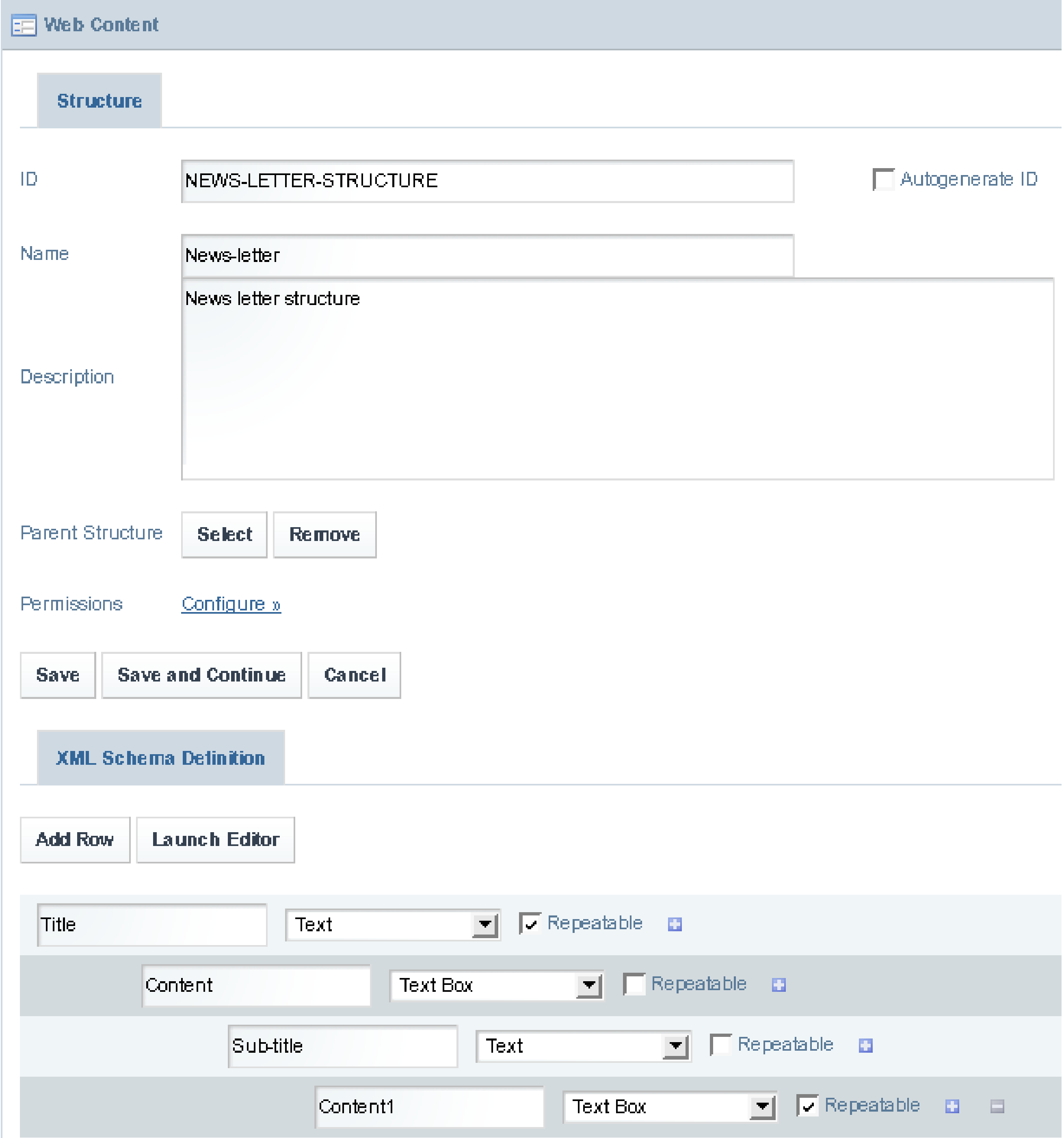
1. **Login to WebSynergy.**

2. **Click Add Applications from the welcome menu.**

3. **Expand the Content Management folder.**

4. **Click Add against the Web Content portlet.**

5. **To create a structure, first click the Structures tab in the Web Content portlet, and then click on the Add Structure button. The structure creation page is displayed.**



ID: NEWS-LETTER-STRUCTURE ☐ Autogenerate ID

Name: News-letter

Description: News letter structure

Parent Structure

Permissions: Configure »

Title | Text | ☑ Repeatable

Content | Text Box | ☐ Repeatable

Sub-title | Text | ☐ Repeatable

Content1 | Text Box | ☑ Repeatable

FIGURE 6–3 To create a Structure

a. **Specify an ID, or Select the *Autogenerate ID* checkbox for the system to generate an ID.**

   It is recommended that you choose your own ID so that the structure will be easier to identify when associating it with templates and articles. The convention is to separate words in the ID with a hyphen as in “NEWS-LETTER-STRUCTURE”. The structure ID can not be changed once your structure has been saved, so be sure that the ID is exactly how you would want it to appear.

b. **Enter a name and a description for the structure.**

c. **Design your XML Schema Definition.**

   You need to define an XML schema for the structure. The *Add Row* button is used to add the XML scheme from the user interface. On clicking the *Add Row* button, a new field where you can enter a name and a content type for the field (image, text, boolean flag, etc.) is displayed. Those comfortable with XML or HTML can edit the XSL directly by clicking *Launch Editor*. Both methods of adding/editing elements will work, and both can be used in the same session. One simple approach is to add a few elements using the GUI and then to edit the XML directly (which will contain the syntax for the already added elements and can be copy-pasted).

   Empty spaces are not allowed within a field name. Rows can be moved up or down by clicking on Arrow buttons. You can use the Plus button to add multiple levels of rows nested inside one another. A field can be repeatable or non-repeatable. Selecting the *Repeatable* option makes a field repeatable. If a field is repeatable, you can repeat the field in an article using the structure. You will be able to add the field as many times as it is desired, by clicking a plus sign.

6 **Click Save.**

The Structure is created successfully. When you create a structure, the system generates a *WebDAV URL* for the structure. WebDAV (Web-based Distributed Authoring and Versioning) makes it possible to remotely access the structure using the WebDAV URL. To determine the WebDAV URL for the structure, open the structure in edit mode after creating it.

For details on WebDAV visit:

http://en.wikipedia.org/wiki/WebDAV

http://www.webdav.org/

▼ To Edit a Structure

1 **Login to WebSynergy.**

2 **Click Add Applications from the welcome menu.**

3 **Expand the Content Management folder.**

4. **Click Add against the Web Content portlet.**

5. **Click the Structures tab.**

6. **Click the Actions button corresponding to a structure, and select Edit from the menu.**

   The Structure opens in Edit mode.

   In the edit screen, you can edit anything in the structure except for the structure ID, which can not be changed once the structure has been created.

   Editing the name and description will not affect other features of the Web Content portlet, but adding, deleting, and modifying fields will affect the templates and articles that are associated with the structure. It is advisable to backup the current XML file first by clicking the *Download* button so that it will be easy to revert to your old structure if necessary.

   If you add fields, the articles associated will have extra fields for additional content. If you delete fields, associated content in associated articles will be deleted.

7. **Make desired changes to the Structure, and click Save to save changes.**

▼ To Delete a Structure

1. **Login to WebSynergy.**

2. **Click Add Applications from the welcome menu.**

3. **Expand the Content Management folder.**

4. **Click Add against the Web Content portlet.**

5. **Click the Structures tab.**

6. **To delete a structure, you must first delete all the articles and disassociate or delete all the templates associated with that structure.**

   To delete all articles, click on the *Actions* button and select *View Articles* from the menu.

   All the articles associated with the Structure are listed.

7. **Select all articles, and click on the Delete button.**

8. **Click the Structure tab again. Click the Actions button corresponding to a Structure, and select View Templates from the menu.**

   All the templates associated with the structure are listed.

You can delete all the Templates associated with the Structure. To do this, select all Templates and then click the *Delete* button.

9 **Select the Structure and click on the Delete button to delete the structure.**

Alternatively, click *Delete* from the menu appearing on clicking the *Actions* button.

Templates

- "To Create a Template" on page 185
- "To Edit a Template" on page 187
- "To Delete a Template" on page 187

▼ To Create a Template

1 **Login to WebSynergy.**

2 **Click Add Applications from the welcome menu.**

3 **Expand the Content Management folder.**

4 **Click Add corresponding with Web Content portlet.**

The *Web Content* portlet is added to your page.

5 **Click the Templates tab.**

Click the *Add Template* button to create a template. The *Template* page appears.

6 **Specify an ID, Name and Description for the template. Click the Select button to select a Structure for the template.**

The *Select* and *Remove* buttons corresponding to the *Structure* field can be used to associate or to disassociate a structure from the Template.

A window for selecting Structures is launched.

7 **Click the name of the Structure to select it.**

For example, click *News-letter*. The structure *News-letter* is selected.

8. **To edit the template layout content, you can use one of two methods:**

   - **To create a completely new template:**

     a. **Click the Launch Editor button.**

        An Editor pops-up. You can select an Editor Type. *Plain* and *Rich* are the two available editor types. Plain is the default editor type, but you can select Rich, as it is easier to work with Rich editor.

     b. **Click Update after making changes to the template.**

   - **To make use of an existing template:**

     a. **Select a language from the Language Type list.**

        VM, *SXL*, and *CSS* are the available language options.

     b. **Click the Browse button to upload the script for an existing template.**

     c. **Select the Format Script checkbox.**

     d. **Click Launch Editor to open the browsed script.**

     e. **Click Update after making changes to the script.**

9. **If you find it to be necessary, Add a small image to represent the template in the description field,**

   You can , enter the URL path for an image in the *Small Image URL* field, or you can browse an image using the *Browse* button. To use the image in the template, select the *Use Small Image* checkbox.

10. **Click Save.**

    The template is created successfully. When you create a template, the system generates a *WebDAV URL* for the template. WebDAV (Web-based Distributed Authoring and Versioning) makes it possible to remotely access the template using the WebDAV URL. To determine the WebDAV URL for the template, open the template in edit mode after creating it.

    For details on WebDAV visit:

    http://en.wikipedia.org/wiki/WebDAV

    http://www.webdav.org/

▼ To Edit a Template

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**
4. **Click Add corresponding with Web Content portlet.**
5. **Click the Templates tab.**
6. **Click the Actions button corresponding to a template, and select Edit from the menu.**

   Before editing the actual template, it is advisable to view the associated articles because their layout will be affected by any changes made to your template. After editing the template, you can view these articles with the new template layout. You can click *View Articles* from the *Action* menu to view associated articles.
7. **Edit the template as desired.**

   In the edit screen, you can edit anything in the template except for the template ID, which can not be changed once the template has been created.
8. **Click Save.**

▼ To Delete a Template

1. **Login to WebSynergy.**
2. **Click Add Applications from the welcome menu.**
3. **Expand the Content Management folder.**
4. **Click Add corresponding with Web Content portlet.**
5. **Click the Templates tab.**
6. **To delete a template, you must first delete all the articles associated with that template.**

   To delete all articles, click the *Actions* button and select *View Articles* from the menu.

   All the articles associated with the Template are listed.

7 **Select all articles, and click Delete.**

A dialog-box with the message "Are you sure you want to delete the selected articles?" appears. Click *OK* to delete all articles.

8 **Click the Templates tab once again.**

9 **Select the template and click *Delete*.**

Click *OK* on the dialog-box for confirming deleting the template.

Articles

- "To Create an Article" on page 188
- "To Edit an Article" on page 190
- "To Delete an Article" on page 190

▼ To Create an Article

1 **Login to WebSynergy.**

2 **Click Add Applications from the welcome menu.**

3 **Expand the Content Management folder.**

4 **Add the Web Content portlet to the page.**

5 **Click the Articles tab.**

All articles are listed.

6 **To create a new article, click the Add Article button.**

You can create a basic or an advanced article.

- **To create a basic article:**

  A basic article do not make use of a template, workflow, and other sophisticated articles.

  a. **Click "<< Basic" in the top-right corner of the portlet window.**

     Skip this step if you are already on this page.

  b. **Create a basic article by providing a name, abstract, and the body of content.**

  c. **Click Save.**

- **To create an advanced article:**

  An advanced article too can be created the same way as you create a basic article, but you can apply *Form and Presentation*, *Workflow*, *Categorization*, and *Schedule* options for the article.

  a. **Click "<< Advanced" in the top-right corner of the portlet window.**

     Skip this step if you are already on this page.

  b. **To select a Structure and a Template for the article, click the Select button besides the Template field.**

     The option to select a structure and a template is found under *Farm and Presentation*.

     A dialog box appears with the message "Selecting a new template will change the structure, available input fields, and available templates? Do you want to proceed?" appears. Click *OK*. The "Templates" pop-up appears. You can select a template by clicking on the name of the template. In our example, click "NEWS-LETTER-TEMPLATE" to select it as the template for the article.

     The structure used by the template is added to the article. You can note that the structure of the article has changed.

     Note – If you want to remove the template or the structure applied to an article, click *Remove* corresponding to the structure.

     ---

  c. **Specify a name for the article.**

  d. **Choose the Type for Categorization that the article will reside in.**

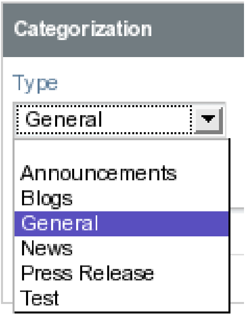


FIGURE 6–4 To create an Article

This makes it easier for users in searching for an article and to organize articles.

e. **Set the Schedule details.**

Set values for *Display Date*, *Expiration Date*, and *Review Date* fields. The default setting does not set an expiration date and review date.

f. **Create the article, and click Save.**

▼ To Edit an Article

1 **Login to WebSynergy.**

2 **Click Add Applications from the welcome menu.**

3 **Expand the Content Management folder.**

4 **Add the Web Content portlet to the page.**

5 **Click the Articles tab.**

All articles are listed.

6 **Click on the Actions button for an article and select Edit from the menu.**

The article appears in edit mode. In this screen you can edit anything in the article except for the article ID, which can not be changed once the article has been created. You can choose to increment the version every time you edit the article to help all users see how often the article has been changed. If you feel that the changes you are making are very minor, you can choose not to increment it. Note that if you set the expiration date to a date earlier to the current date, you will no longer be able to select that article for use in the *Web Content Display* portlet.

7 **Make necessary changes and click Save.**

▼ To Delete an Article

1 **Login to WebSynergy.**

2 **Click Add Applications from the welcome menu.**

3 **Expand the Content Management folder.**

4 **Add the Web Content portlet to the page.**

5 **Click the Articles tab.**

All articles are listed.

6 **Click on the Actions button for an article and select Delete from the menu.**

Alternatively, select the checkbox preceding the article ID and click the *Delete* button.

Web Content List Portlet

Admin users can configure the Web Content List portlet to display the articles associated with a community. You can have more than one instance of this portlet on a page.

▼ To Configure the Journal Articles Portlet

1 **Login to Project WebSynergy as Admin User.**

2 **Click Add Applications from the welcome menu.**

3 **Add the Web Content List portlet to your page.**

4 **Choose the Configuration button from the menu controls for the portlet.**

5 **Choose Current tab under Setup tab, and select values for Community, Article Type and other attributes.**

If you are using WebSynergy samples, you can select 'My Community' and 'General' for Community and Article Type respectively.

Note – The Community *My Community* is the default community, if you are using WebSynergy samples.

---

6 **Click Save, and click the Return to Full Page link.**

The portlet lists the articles available on the Community 'My Community'.

Web Content Display Portlet

You can create, configure, and display articles using the Web Content Display portlet. You can have more than one instance of this portlet on a page.

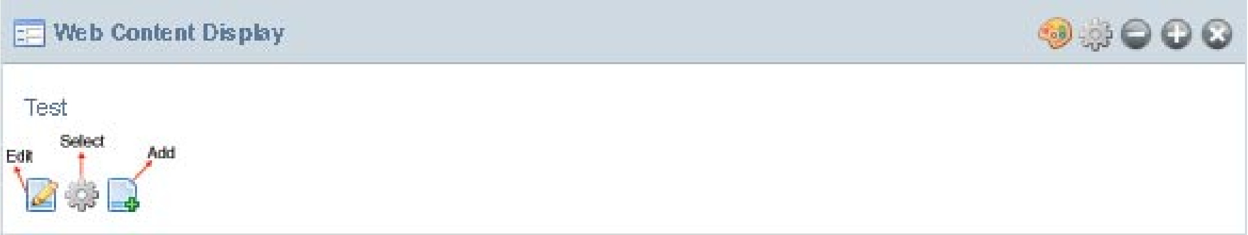


FIGURE 6–5 Web Content Display Portlet

The portlet embeds **Edit Article**, **Select Article**, and **Add Article** icons, respectively for editing an article selected for display, for selecting an article for display, and for adding a new article.

Edit Article This icon provides the interface to edit the article displayed using the portlet.

Select Article This icon provides the interface to select an article for display.

You can enable a number of settings on the articles which are displayed. Enabling printing, comments, and comment ratings are some of the settings which can be applied to an article which is displayed.

Add Article This icon provides the interface to create a new article for display.

▼ To Select an Article for Display

1. **Login to Project WebSynergy as Admin User.**

2. **Click Add Applications from the welcome menu.**

3. **Add the Web Content Display portlet to your page.**

4. **Click the Select Article icon (which is the same as the Configuration button).**

5. **Click Search Articles.**

   All available articles are displayed.

6. **Click on an article to select it for display.**

7. **Apply the Setup and Supported Client configurations.**

   - **Configure the display setup by selecting the options available under the Setup tab.**
     - **Show Available Locales**

       Enable this option to display all the locales which are available for the article.
     - **Convert To**

Enable this option to support converting the article to different document formats.

- **Enable Print**

  Enable this option to make it possible to print the article.
- **Enable Ratings**

  Enable this option to make it possible to rate the article.
- **Enable Comments**

  Enable this option to make it possible to add comments for the article.
- **Enable Comment Ratings**

  Enable this option to make it possible rate the comments received for the article.

- **To configure the supported clients:**

  a. **Open the Supported Clients tab.**

  The *Mobile Devices* option is enabled by default. Disable the option, in case it is not required to provide display support for mobile devices.

  b. **Click Save.**

8 **Click Return to Full Screen.**

Web Content Search Portlet

The Web Content Search portlet can be used to search for a word or phrase within the content of an Article.

You can define the search behavior by making changes to the search configuration.

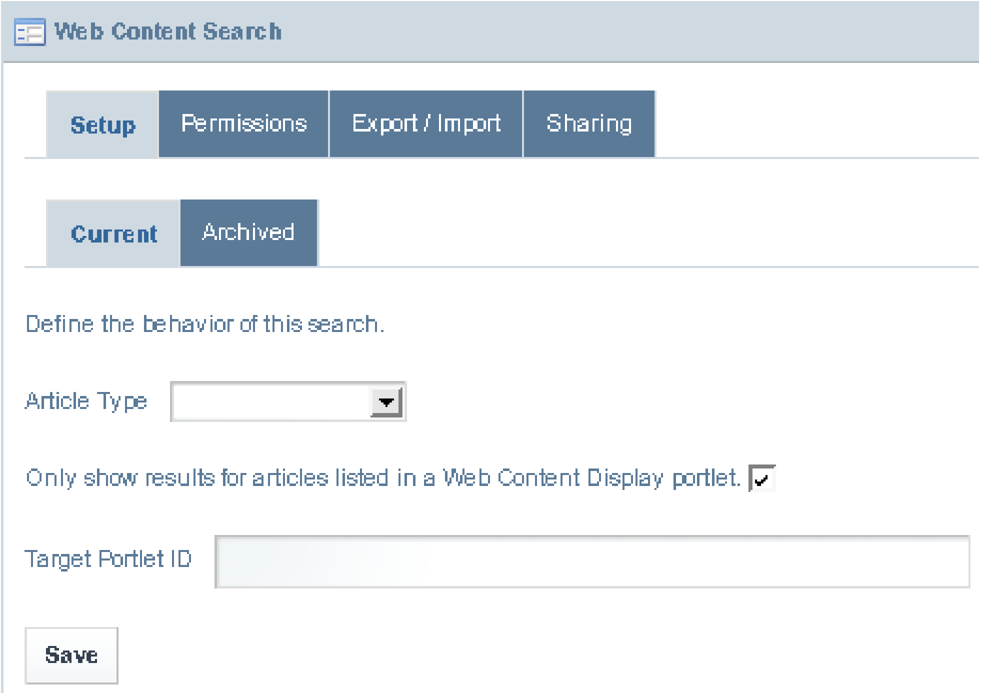
▼ To Customize the Search Behavior

1 **Login to Project WebSynergy as Admin User.**

2 **Click Add Applications from the welcome menu.**

3 **Add the Web Content Search portlet to your page.**

4 **Click the Configuration button from among menu controls.**



Web Content Search

Define the behavior of this search.

Article Type

Only show results for articles listed in a Web Content Display portlet. ☑

Target Portlet ID

FIGURE 6–6 Configuring the search behavior

5. **Define the search behavior using:**

   - **Article Type**

     Select the article type which need to be included in the search. The default search includes all types of articles. When you select an article type, search includes only the articles of that type.

   - **Only show results for articles listed in a Web Content Display portlet**

     By default, search includes all the articles which are listed in any of the instances of the Web Content Display portlet.

   - **Target Portlet ID**

     You can specify the instance ID of the Web Content Display portlet to conduct search only inside that particular instance of the portlet. You need to disable the “Only show results for articles listed in a Web Content Display portlet” option before you specify the target portlet instance ID.

6. **Click Save.**

Asset Publisher Portlet

To be done.

▼ To add Asset Publisher Portlet to the Page

1 **Login to WebSynergy.**

2 **Click Add Applications and expand the Content Management folder.**

3 **Add the Asset Publisher portlet to your page.**

The *Asset Publisher* portlet is added to your page.

Blogs Entry

To be done.

▼ To add a Blogs Entry

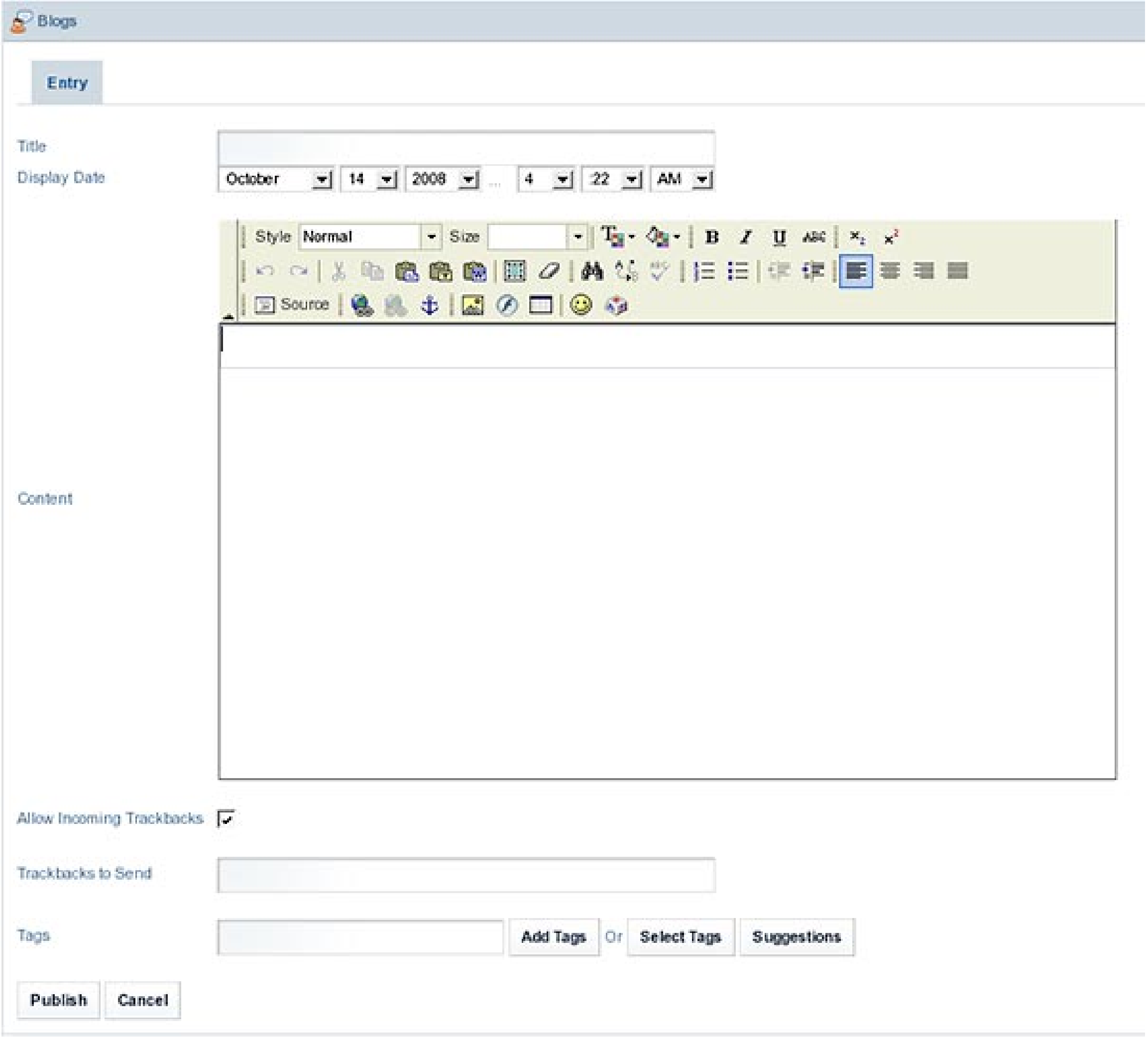
1 **Login to WebSynergy.**

2 **Click Add Applications and expand the Content Management folder.**

3 **Add the Asset Publisher portlet to your page.**

The *Asset Publisher* portlet is added to your page.

4 **Select Blogs Entry from the list on the Asset Publisher portlet.**



Display Date: October 14 2008 4 22 AM

Style: Normal

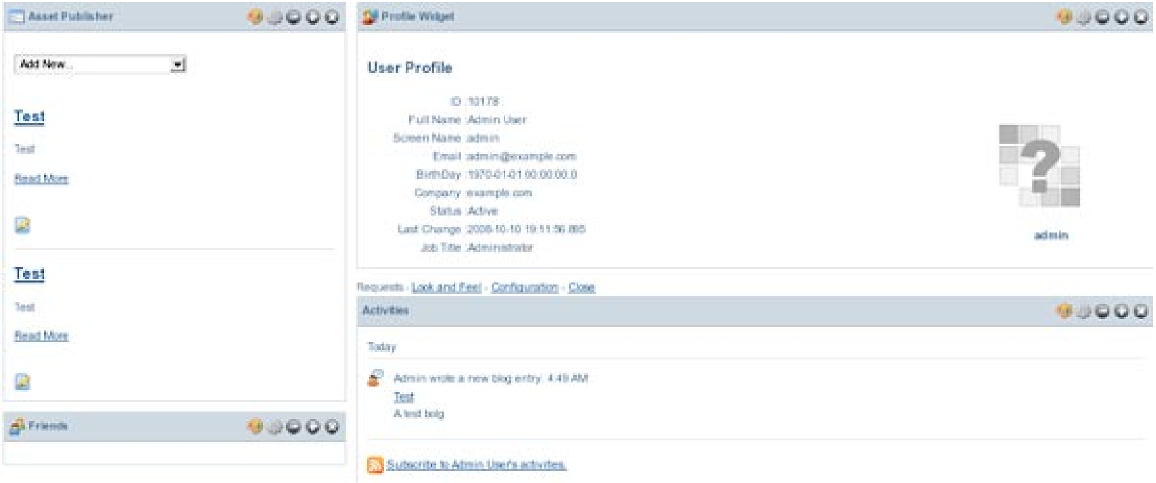
Allow Incoming Trackbacks ☑

FIGURE 6–7 Page for entering the blog

The Blogs Entry page has many sophisticated options. You can apply styles and formatting on your blog entry. You can add Tags and Suggestions to your blog

5 **Make your blog entry and click Publish.**

The blog entries made by you are displayed on the *Asset Publisher* portlet. For example, you have made two blog entries with the name *Test*. The two blog entries are displayed on the Asset Publisher. The *Activities* portlet also states that a blog entry has been made.



Test

Test

Read More

Test

Test

Read More

User Profile

ID: 10178
Full Name: Admin User
Screen Name: admin
Email: admin@example.com
BirthDay: 1970-01-01 00:00:00.0
Company: example.com
Status: Active
Last Change: 2008-10-10 19:11:56.895
Job Title: Administrator

admin

Requests - Look and Feel - Configuration - Close

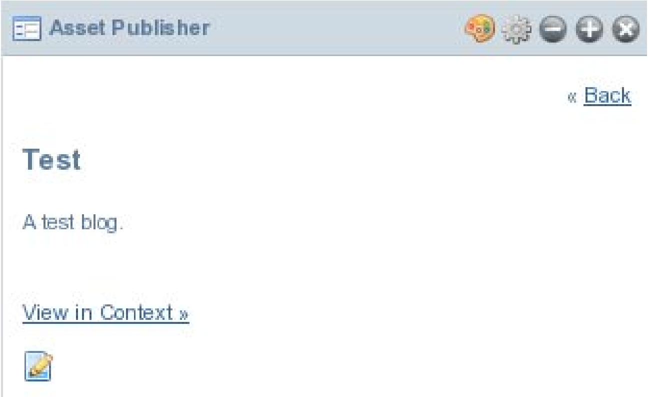
Today

Admin wrote a new blog entry. 4:49 AM
Test
A test bolg

Subscribe to Admin User's activities.

FIGURE 6–8 Blog entries

6. **To view the blog entry, click on the blog title or on Read More.**



« Back

Test

A test blog.

View in Context »

FIGURE 6–9 Viewing the blog

7. **You can edit the blog entry by clicking on the Edit button.**

Bookmarks Entry

To be done.

▼ To add a Bookmarks Entry

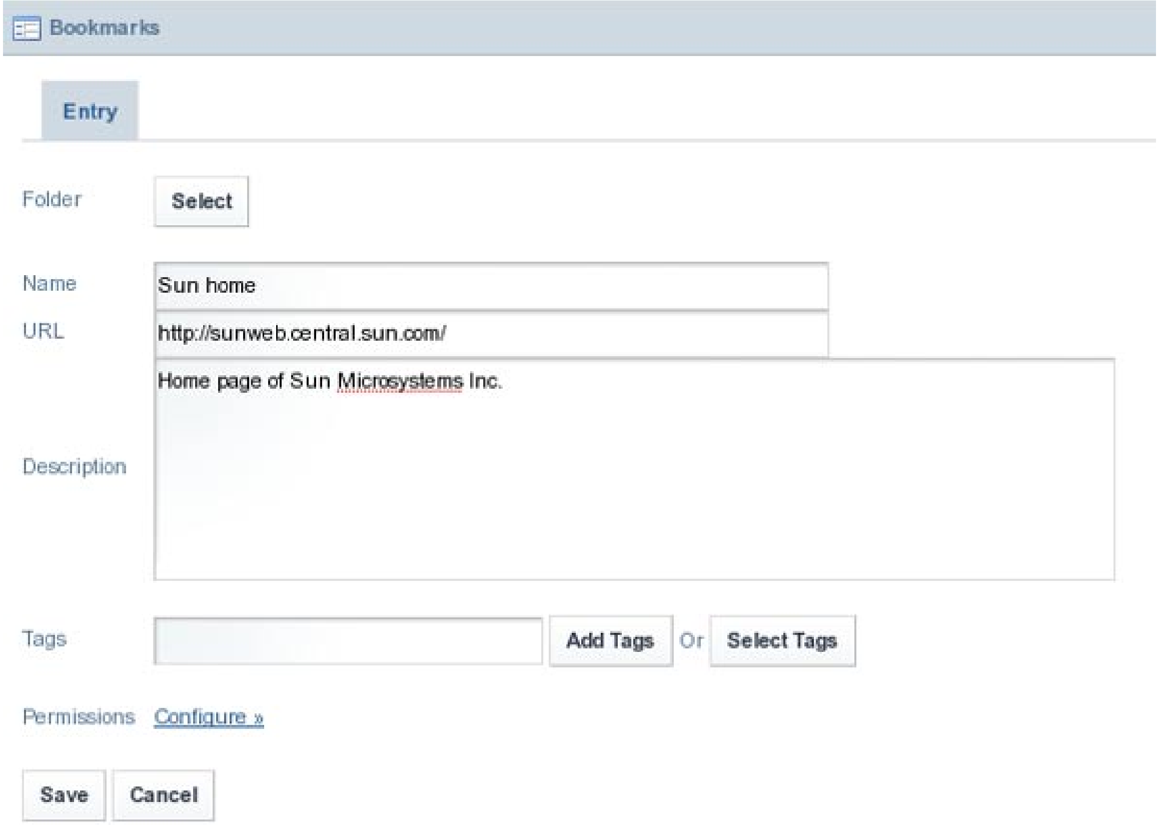
1 **Login to WebSynergy.**

2 **Click Add Applications and expand the Content Management folder.**

3 **Add the Asset Publisher portlet to your page.**

The *Asset Publisher* portlet is added to your page.

4 **Select Bookmarks Entry from the list on the Asset Publisher portlet.**



Name: Sun home

URL: http://sunweb.central.sun.com/

Description: Home page of Sun Microsystems Inc.

FIGURE 6–10 Entering a bookmark

5 **Enter the bookmark**

6 Click Select to select a folder.

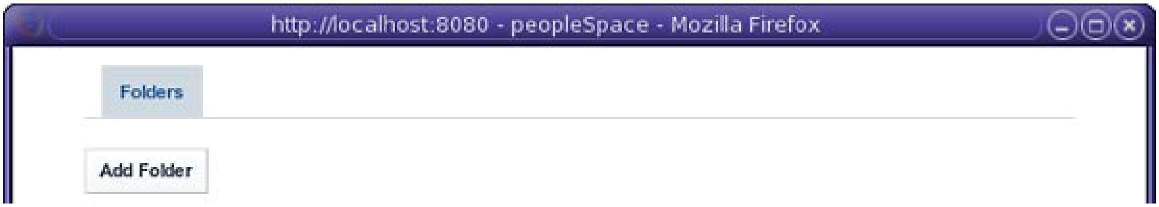
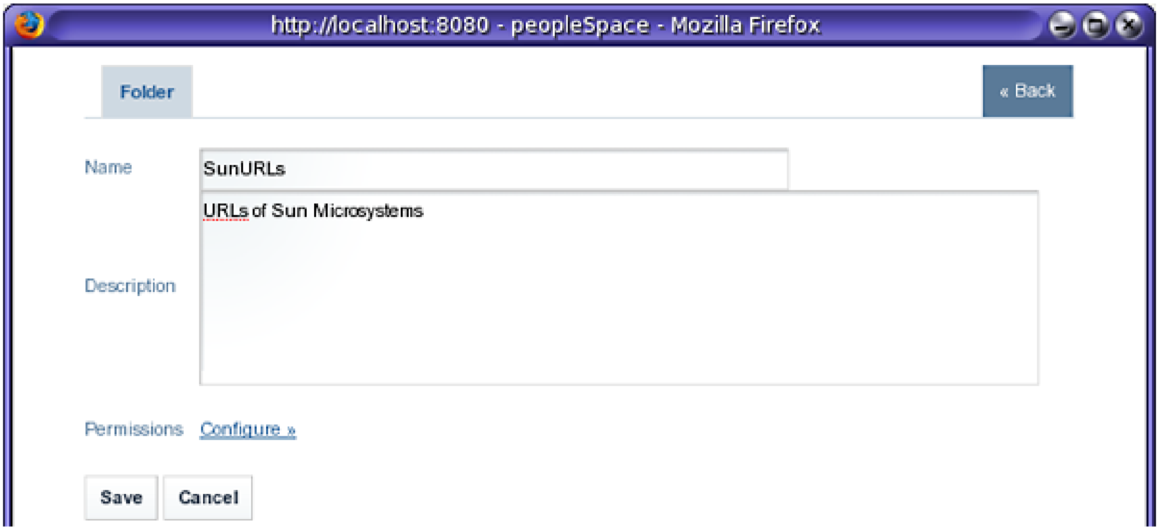


FIGURE 6–11 Selecting a folder

7 Click Add Folder.

8 Specify the name and description of the folder.

Click *Configure* against *Permissions* to view and change the configuration permissions for the folder.

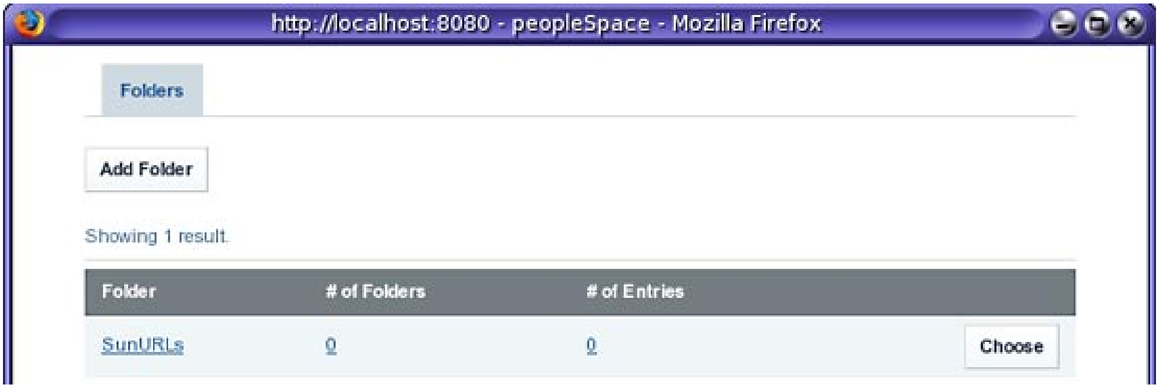


Name: SunURLs

Description: URLs of Sun Microsystems

FIGURE 6–12 Adding a folder

9 Click save to create the folder.

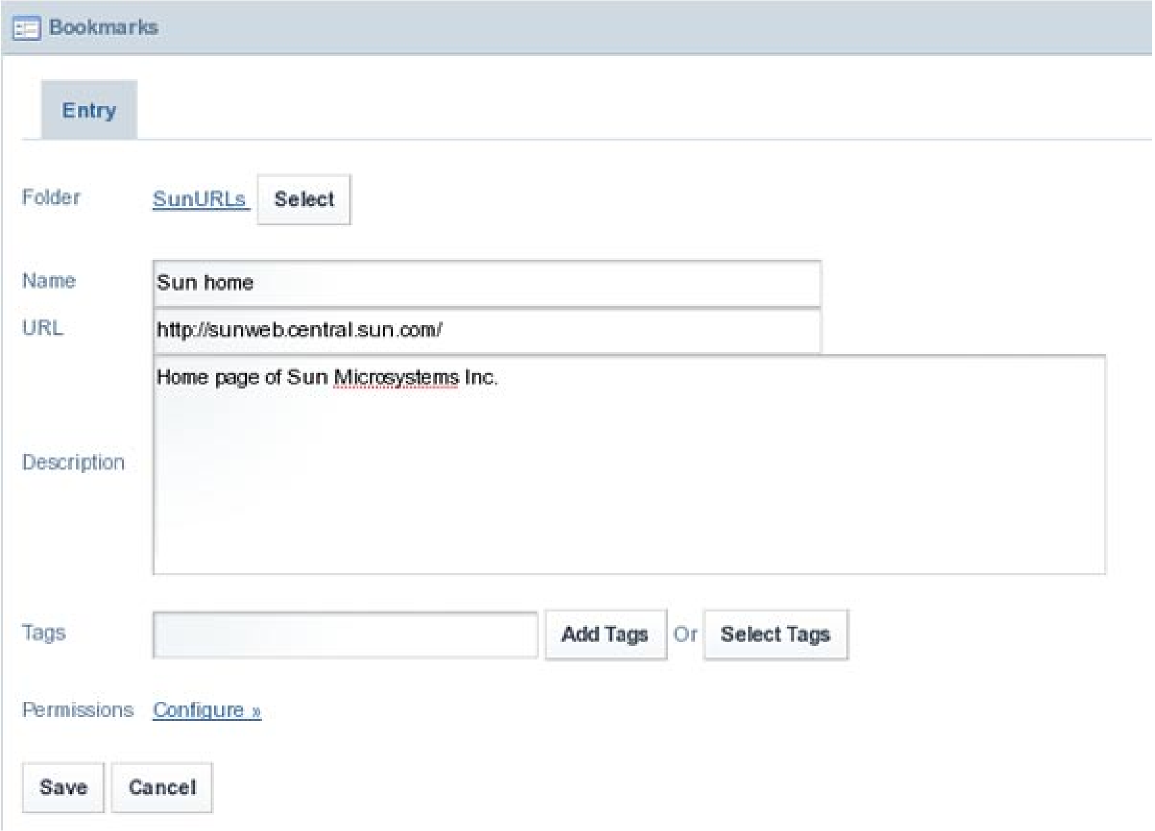


Showing 1 result.

Folder	# of Folders	# of Entries	
SunURLs	0	0	Choose

FIGURE 6–13 Choosing a folder for the bookmark

10 Click on the Choose button to choose the folder.



Folder SunURLs

Name: Sun home

URL: http://sunweb.central.sun.com/

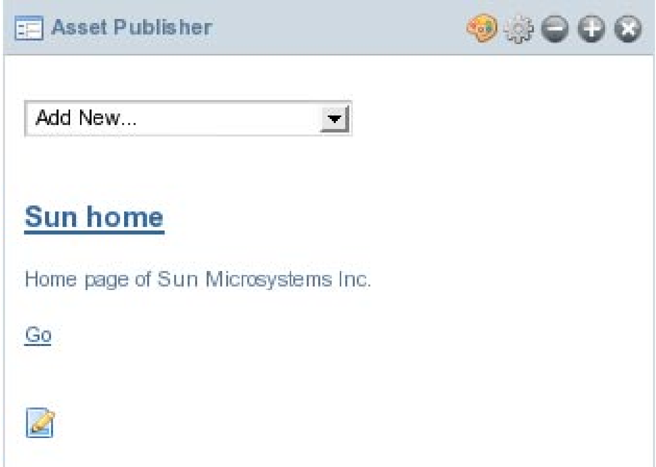
Description: Home page of Sun Microsystems Inc.

Tags

FIGURE 6–14 Saving the bookmark

11 **Click Save.**

The bookmark appears on the Asset Publisher portlet.



Asset Publisher

Add New...

Sun home

Home page of Sun Microsystems Inc.

Go

FIGURE 6–15 Bookmark for a page

12 **Click Go to view the bookmarked page.**

Document Library Document

To be done.

▼ To add a Document Library Document

1 **Login to WebSynergy.**

2 **Click Add Applications and expand the Content Management folder.**

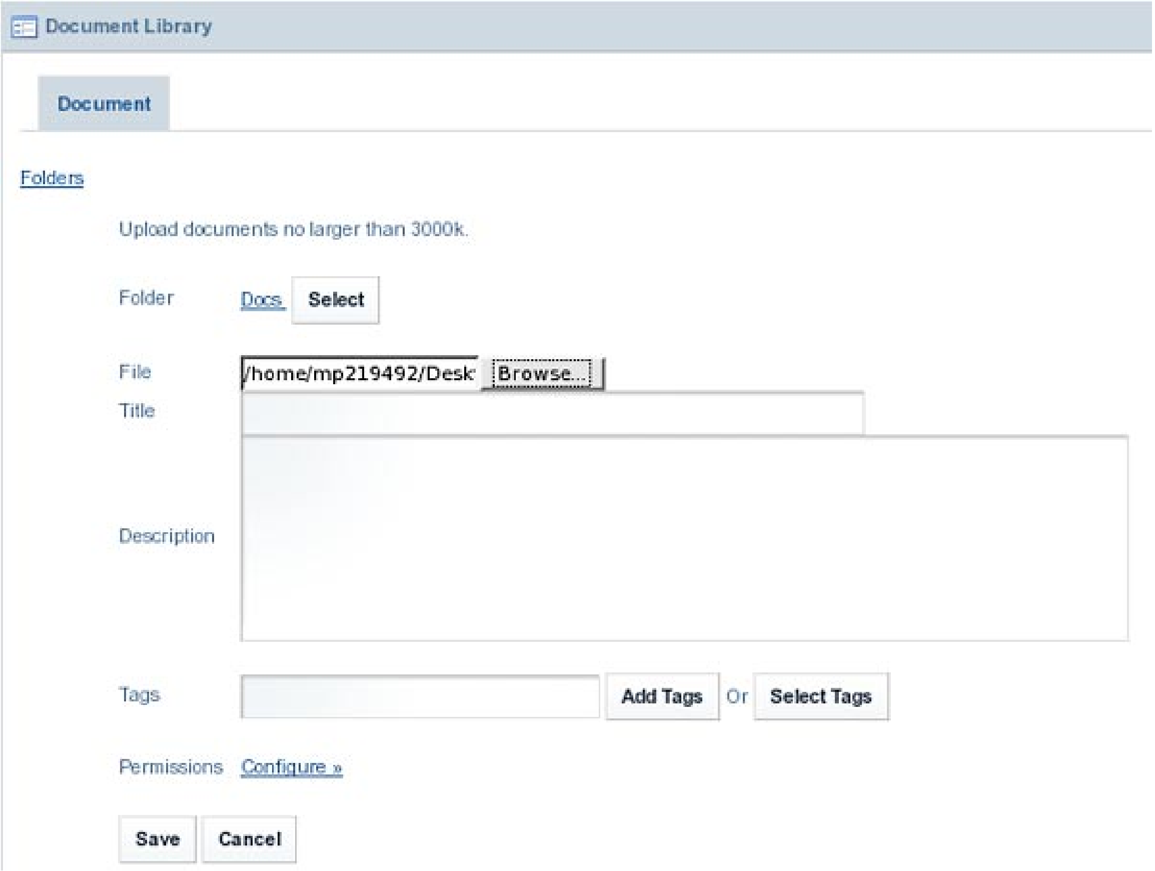
3 **Add the Asset Publisher portlet to your page.**

The *Asset Publisher* portlet is added to your page.

4 **Select Document Library Document from the list on the Asset Publisher portlet.**

5 **Click Select to select a folder as described in the procedure and browse a document to upload it to the Document Library.**

In our example, we are creating a folder called *Docs*, and uploading a file named *Wedding Invitation*.



Document Library

Upload documents no larger than 3000k.

Folder: Docs

File: /home/mp219492/Desk

Title

Description

Tags

Permissions: Configure »

FIGURE 6–16 Adding a Document Library Document

6 **Click Save to save the document to the Document Library.**

The document is displayed on the Asset Publisher portlet. You can download the document if required.

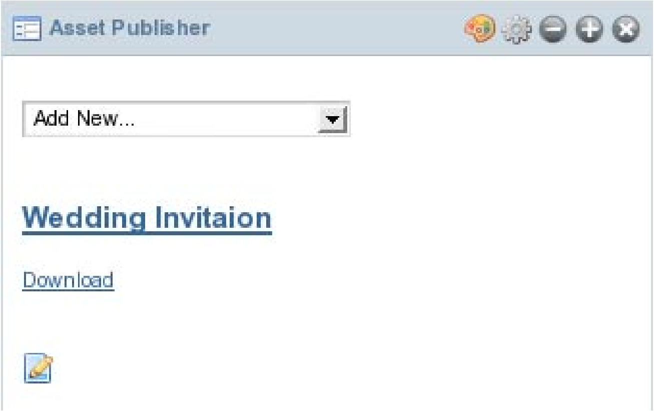


FIGURE 6–17 Saving the document to the Document Library

▼ To Delete a Document from the Document Library

1 **Login to WebSynergy.**

2 **Click Add Applications and expand the Content Management folder.**

3 **Add the Asset Publisher portlet to your page.**

The *Asset Publisher* portlet is added to your page.

4 **Select Document Library Document from the list on the Asset Publisher portlet.**

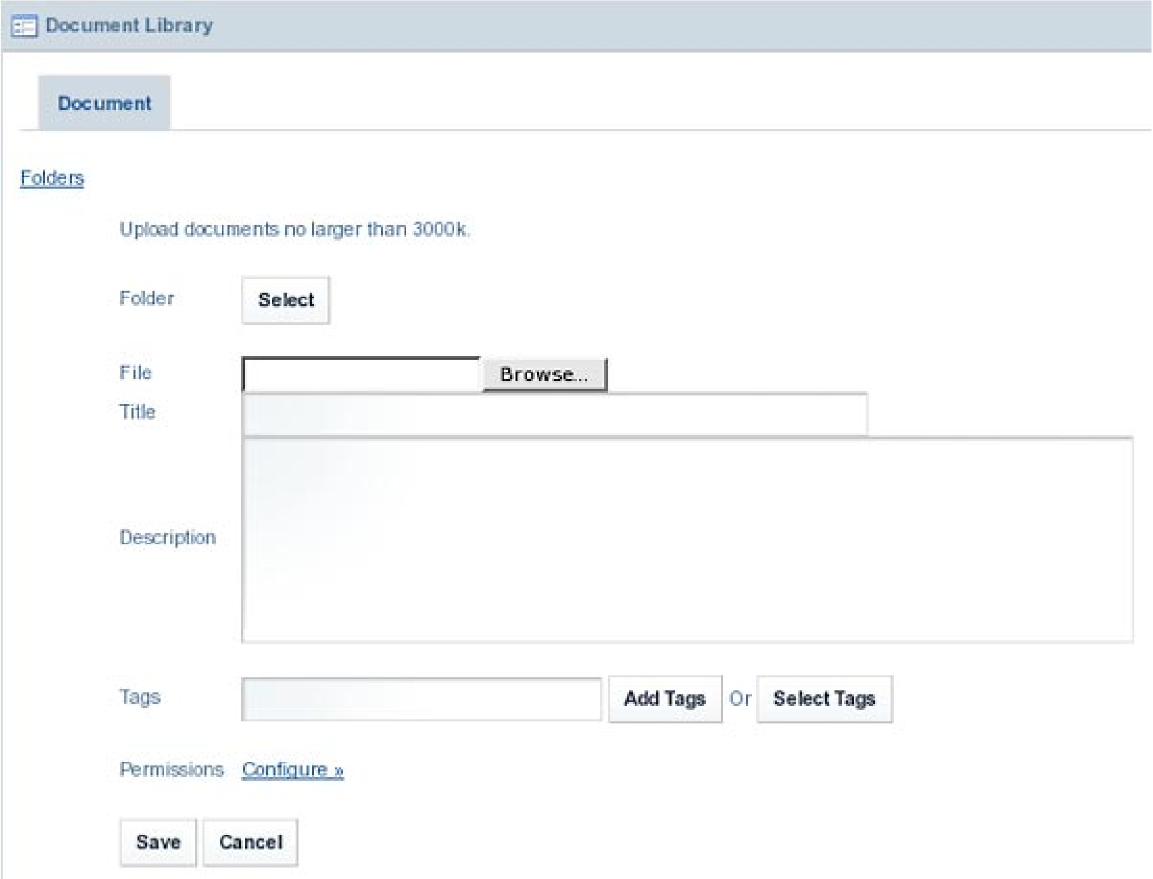


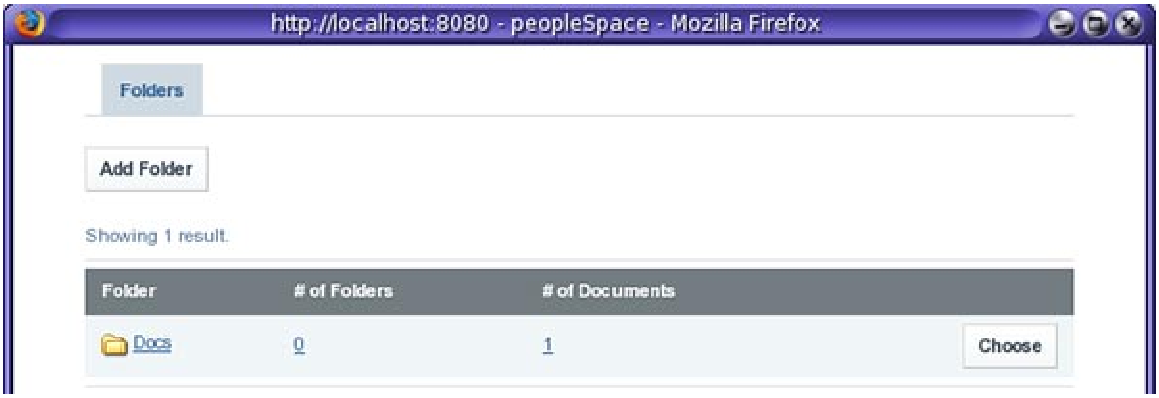
FIGURE 6–18 Document Library page

5. **Click Select.**

   All the documentation folders currently available are listed

6. **Click Choose against the folder containing the document.**

   In our example, choose the *Docs* folder.

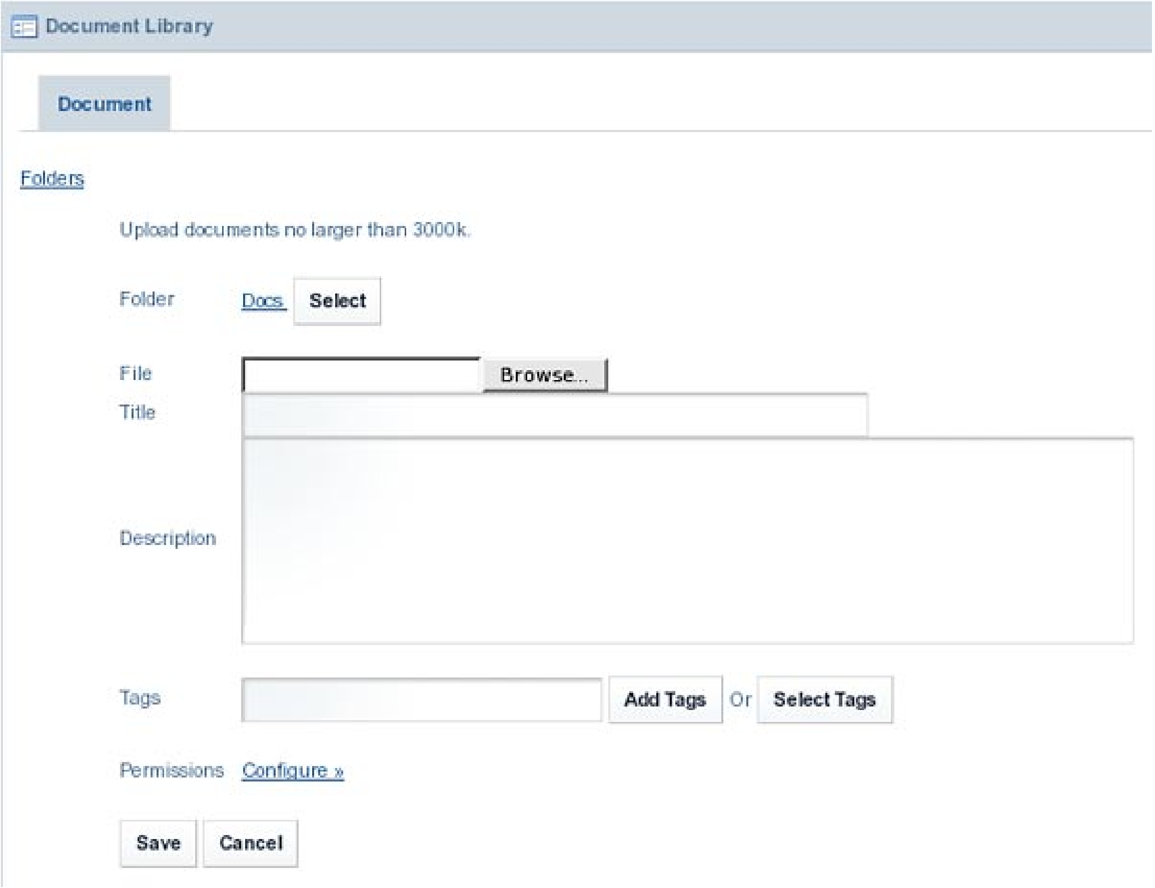


Showing 1 result.

Folder	# of Folders	# of Documents	
Docs	0	1	Choose

FIGURE 6–19 Choosing a documentation folder

The selected folder is added to the Document Library page. Note that a link for the folder appears near the *Select* button.



Document Library

Document

Folders

Upload documents no larger than 3000k.

Folder Docs Select

File

Title

Description

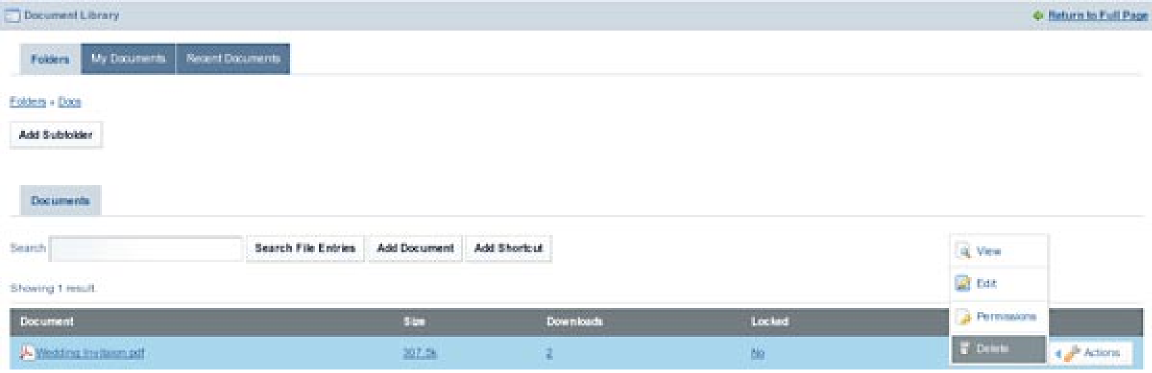
Tags

Permissions Configure »

FIGURE 6–20 Selected folder (containing the document which need to be deleted)

7 **Click on the link for the selected folder.**

8 **Select Delete from Actions, to delete the document.**



Showing 1 result.

Document	Size	Downloads	Locked
Wedding Invitation.pdf	307.5k	2	No

FIGURE 6–21 Deleting the document

Image Gallery Image

To be done.

To add an Image Gallery Image

1. **Login to WebSynergy.**
2. **Click Add Applications and expand the Content Management folder.**
3. **Add the Asset Publisher portlet to your page.**

   The *Asset Publisher* portlet is added to your page.
4. **Select Image Gallery Image from the list on the Asset Publisher portlet.**
5. **The procedure to upload and delete images from the gallery are very much similar to the procedure to To add a Document Library Document and To delete a document from the Document Library respectively.**

Journal Article

To be done.

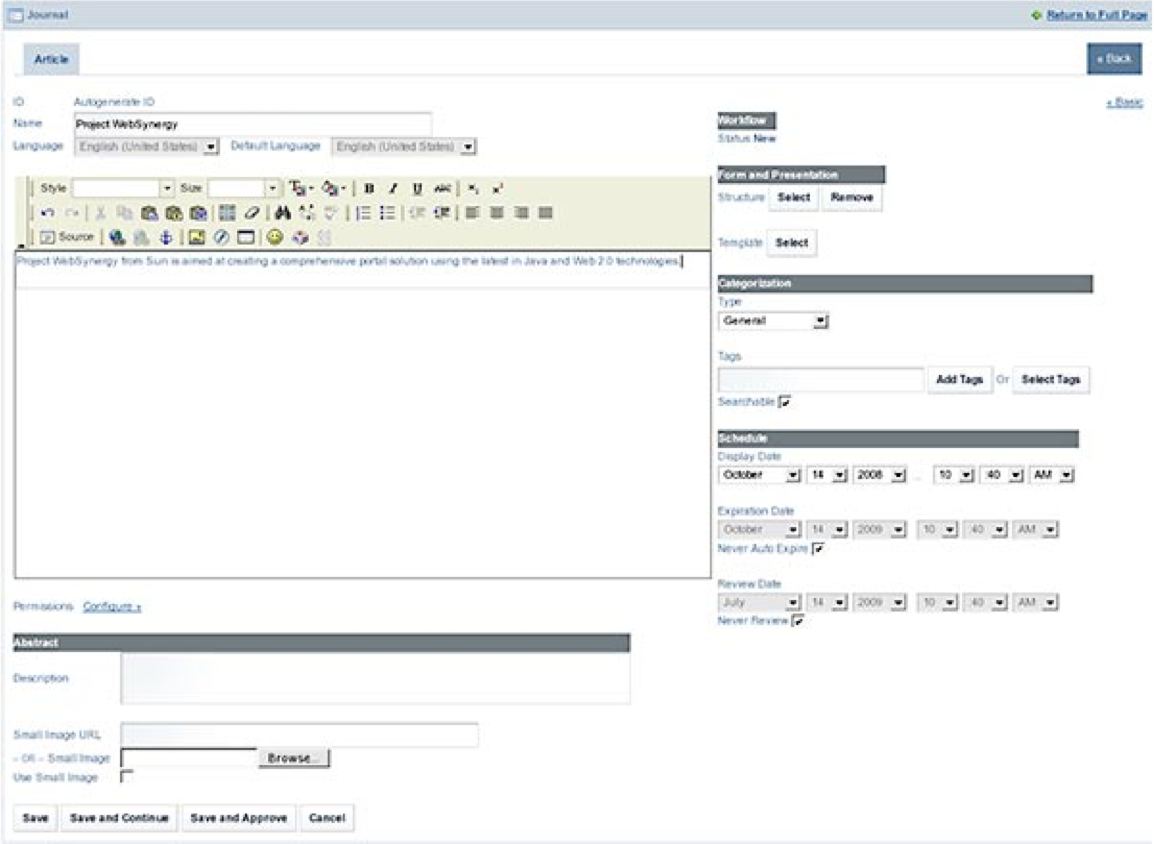
To add a Journal Article

1. **Login to WebSynergy.**
2. **Click Add Applications and expand the Content Management folder.**

3 **Add the Asset Publisher portlet to your page.**

The *Asset Publisher* portlet is added to your page.

4 **Select Journal Article from the list on the Asset Publisher portlet.**



ID Autogenerate ID
Name Project WebSynergy
Language English (United States) Default Language English (United States)

Project WebSynergy from Sun is aimed at creating a comprehensive portal solution using the latest in Java and Web 2.0 technologies.

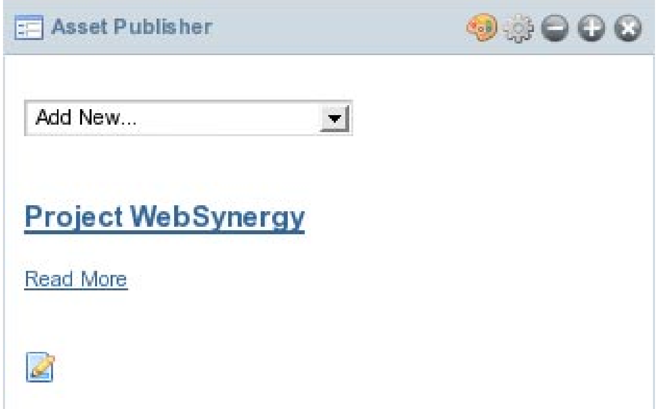
Status New
Structure
Template
Type General
Tags
Searchable ☑
Display Date October 14 2008 10 40 AM
Expiration Date October 14 2009 10 40 AM
Never Auto Expire ☑
Review Date July 14 2009 10 40 AM
Never Review ☑
Description
Small Image URL
– Or – Small Image
Use Small Image □

FIGURE 6–22 Adding a Journal Article

You can create a journal article using all the options available on the screen. You can set an *Expiration Date* and *Review Date* for the journal article.

5 **Click Save and Approve.**

The journal article is displayed on the Asset Publisher portlet. In our example, *Project WebSynergy* is the name of the journal article.



Asset Publisher

Add New...

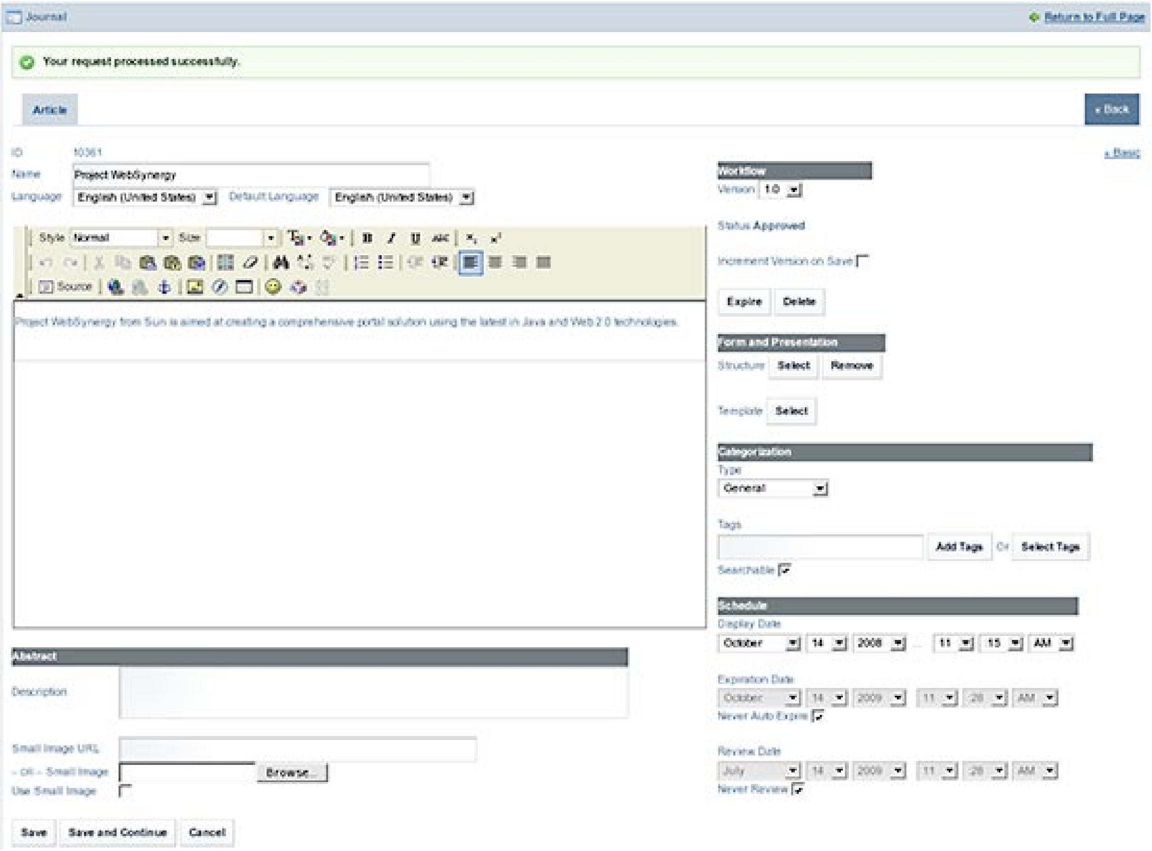
Project WebSynergy

Read More

FIGURE 6–23 Journal Article visible on the Asset Publisher

6 **Click the edit button to make changes to the journal article.**

The default version number of the Journal Article is set to *1.0*. You can select the *Increment Version on Save* option to increment the version number of the article, each time you update the article. You can use the *Expire* button to make the article expire immediately. You can use the *Delete* button to delete the article.



Your request processed successfully.

ID 10361

Name: Project WebSynergy

Language: English (United States) Default Language: English (United States)

Project WebSynergy from Sun is aimed at creating a comprehensive portal solution using the latest in Java and Web 2.0 technologies.

Version 1.0

Status Approved

Increment Version on Save □

Type: General

Searchable ☑

Display Date: October 14 2008 11 15 AM

Expiration Date: October 14 2009 11 28 AM

Never Auto Expire ☑

Review Date: July 14 2009 11 28 AM

Never Review ☑

Use Small Image □

FIGURE 6–24 Editing the Journal Article

7. **Click Save to save changes to the article.**

8. **Click Read More link for the journal article to read the article.**

Breadcrumb Portlet

It displays the path for the page which you are on.

Categories Navigation Portlet

To be done.

It displays an empty portlet. Do we really have anything to document???

Navigation Portlet

To be done.

It displays an empty portlet. Do we really have anything to document???

Nested Portlets Portlet

To be done.

It displays an empty portlet. Do we really have anything to document???

Recent Documents Portlet

It displays the most recent documents.

Site Map Portlet

The site map portlet provides the site mapping of your page.

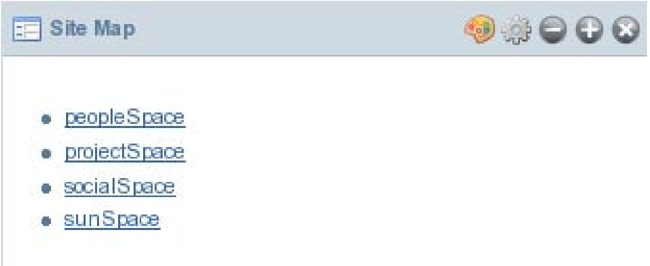


FIGURE 6–25 Site Map Portlet

Tags Admin Portlet

To be done.

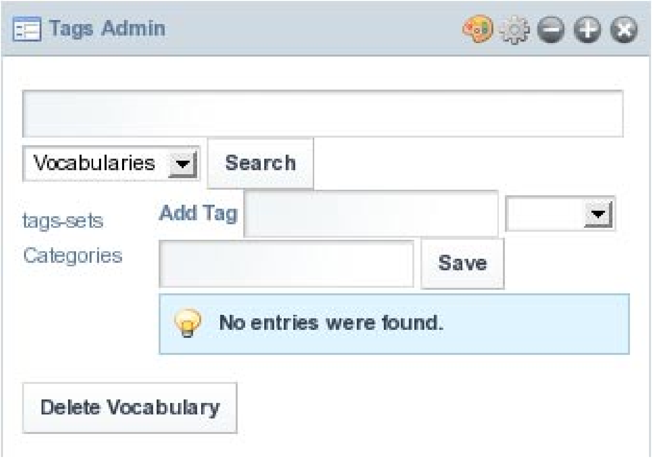
▼ To add Tags Admin portlet to the Page

1. **Login to WebSynergy.**

2. **Click Add Applications and expand the Content Management folder.**

3. **Click Add against the Tags Admin Portlet.**

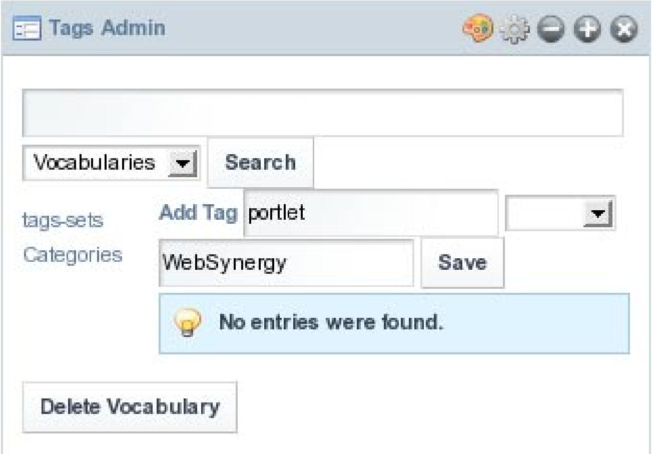
   The Tabs Admin portlet is added to your page.



No entries were found.

FIGURE 6–26 Tags Admin Portlet

4. **Add a tag and a tag category and click Save.**



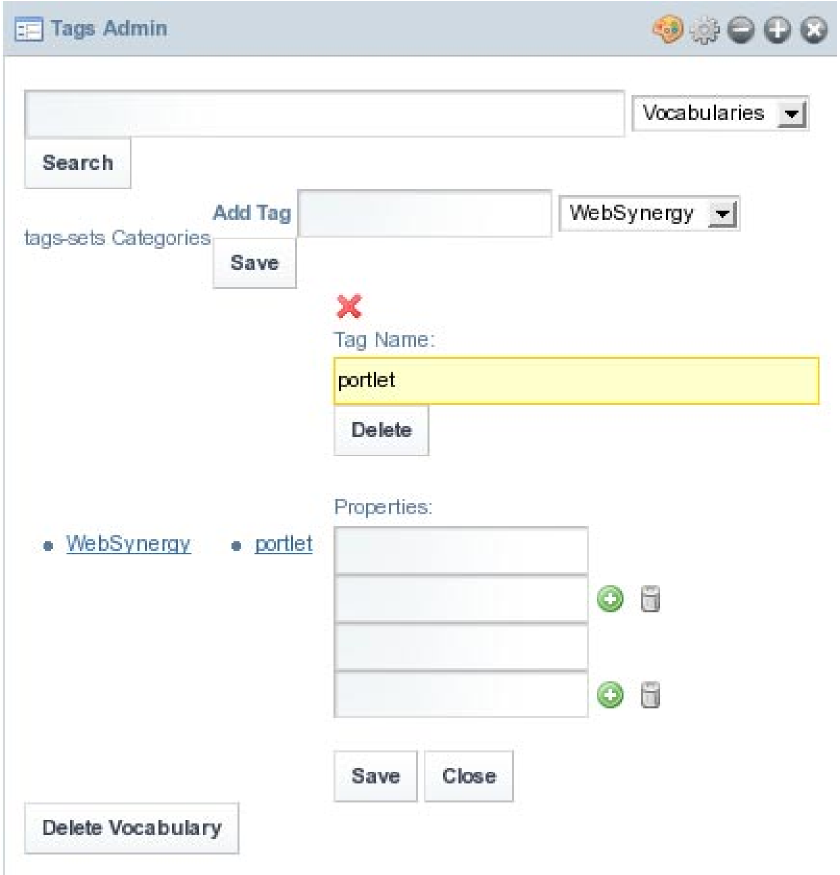
Add Tag portlet

WebSynergy

No entries were found.

FIGURE 6–27 Adding a tag and a tag category

The tag and the tag category are added to the vocabulary.



Vocabularies

WebSynergy

Tag Name: portlet

FIGURE 6–28 Tag vocabulary

You can Add or Delete the properties associated with the tag.

XSL Content Portlet

To be done.

Workflow Portlet

Admin users can define a workflow and upload the workflow definition file through the Workflow Portlet UI, and can initiate a workflow instance. To create a workflow, you need a Workflow Definition File which has a workflow defined in it.

To create a workflow, you need to create a Workflow Definition File in XML format. The Workflow Definition File need to confirm to *jBPM Process Definition Language (JPDL)* standards.

For example, consider that we are creating a workflow for leave application and approval process.

The following is the scenario involved:

1. The admin user creates a workflow definition called LeaveApproval.
2. The admin user creates a workflow instance. (Workflow State 1)
3. The test user uses the workflow instance to apply for leave. (Workflow State 2)
4. The manager receives the holiday request. The manager the approve or reject the request, or can send the request for reconsideration. (Workflow State 2)
5. The test user can verify to see that the process have ended.

Refer to the below sections for detailed process steps:

1. "To add a Workflow Definition" on page 216
2. "To Create a Workflow Instance" on page 219
3. "Workflow Process Steps: To Request an Holiday" on page 221
4. "Workflow Process Steps: To Evaluate Holiday Request" on page 222
5. "Workflow Process Steps: To Ascertain the Request Status" on page 222

▼ To add a Workflow Definition

1. **Login to WebSynergy as Admin User.**

2. **Click on Add Page.**

3. **Specify a name for the page and click Save.**

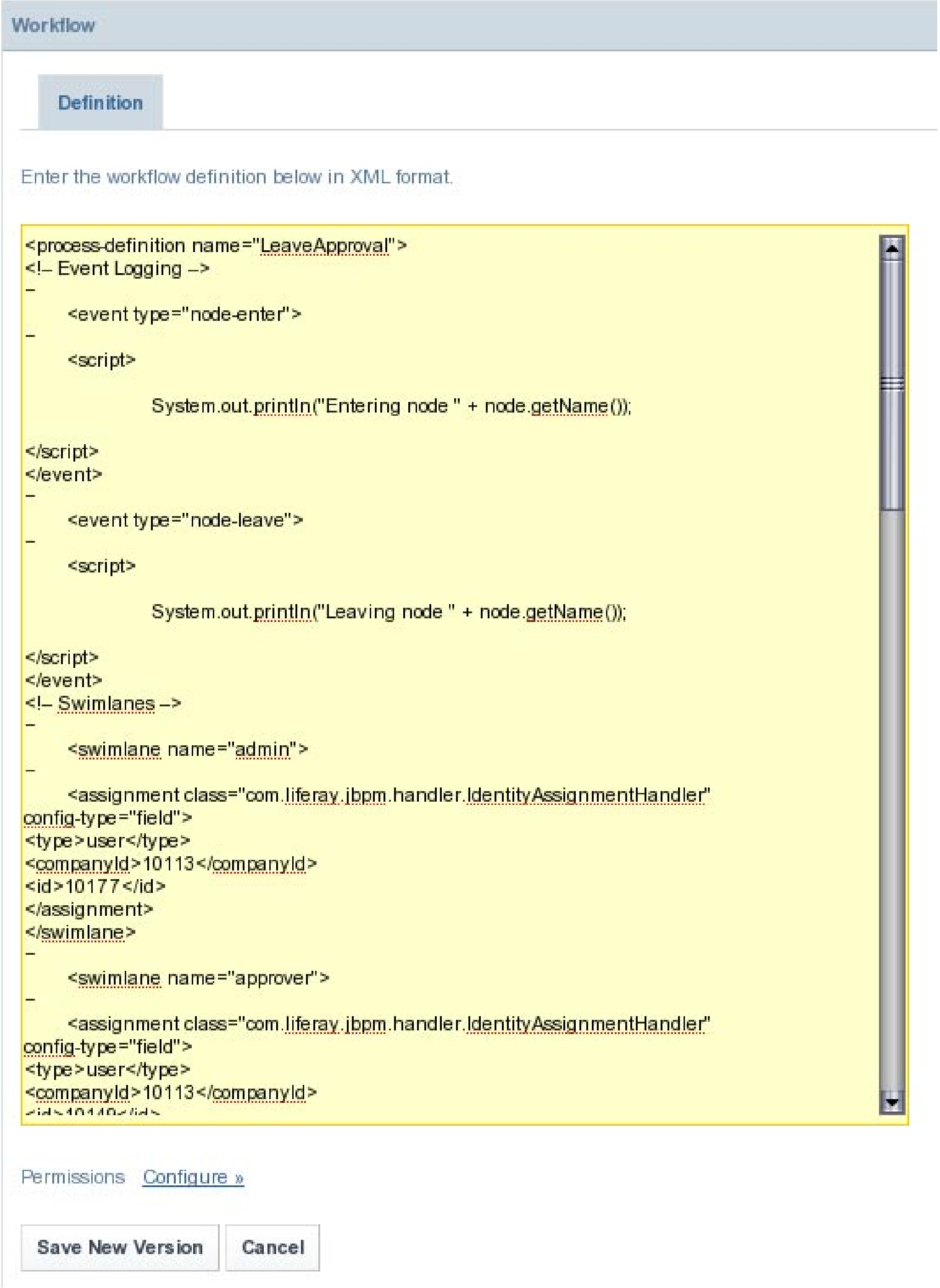
4. **Click on the tab for the page.**

   The page opens.

5. **Click on Add Applications from the Welcome menu, and add the Workflow portlet to the page.**

6. **Click On Add Definition.**

7 **Copy paste the workflow definition in XML format and click on the Save New Version button.**



Enter the workflow definition below in XML format.

```
<process-definition name="LeaveApproval">
<!– Event Logging –>
–
      <event type="node-enter">
–
      <script>

                  System.out.println("Entering node " + node.getName());

</script>
</event>
–
      <event type="node-leave">
–
      <script>

                  System.out.println("Leaving node " + node.getName());

</script>
</event>
<!– Swimlanes –>
–
      <swimlane name="admin">
–
      <assignment class="com.liferay.jbpm.handler.IdentityAssignmentHandler"
config-type="field">
<type>user</type>
<companyId>10113</companyId>
<id>10177</id>
</assignment>
</swimlane>
–
      <swimlane name="approver">
–
      <assignment class="com.liferay.jbpm.handler.IdentityAssignmentHandler"
config-type="field">
<type>user</type>
<companyId>10113</companyId>
<id>10140</id>
```

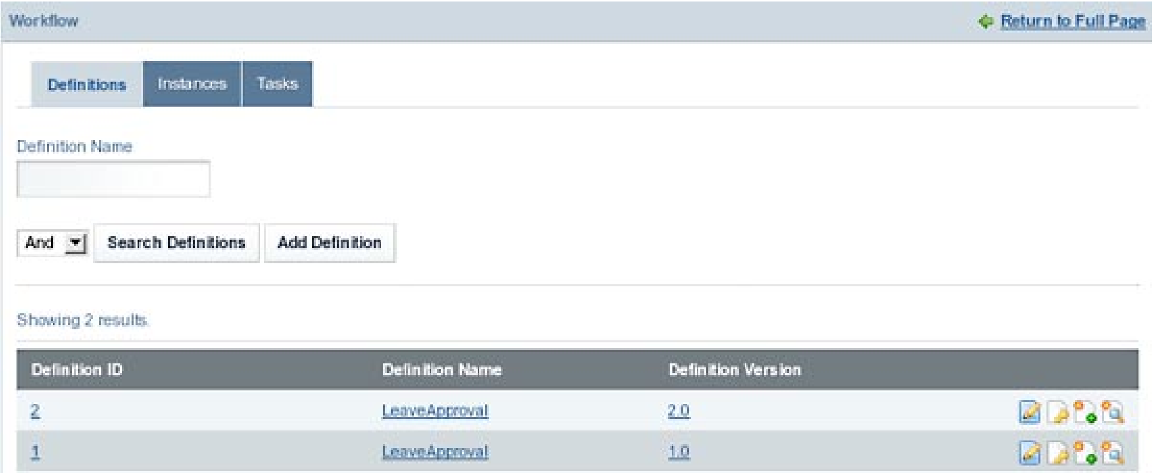
FIGURE 6–29 To add a Workflow Definition

A workflow is created with a version number associated with it.

8 **Logout and login back into WebSynergy for the changes to take effect.**

▼ To Create a Workflow Instance

1 **Click on Search Definitions from the Workflow portlet to view all workflow definitions.**



Showing 2 results

Definition ID	Definition Name	Definition Version
2	LeaveApproval	2.0
1	LeaveApproval	1.0

FIGURE 6–30 To Create a Workflow Instance

2 **Click on the Add Instances button against a workflow definition.**

In our example, we are creating an instance for version *1.0* of the *LeaveApproval* workflow definition.

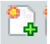
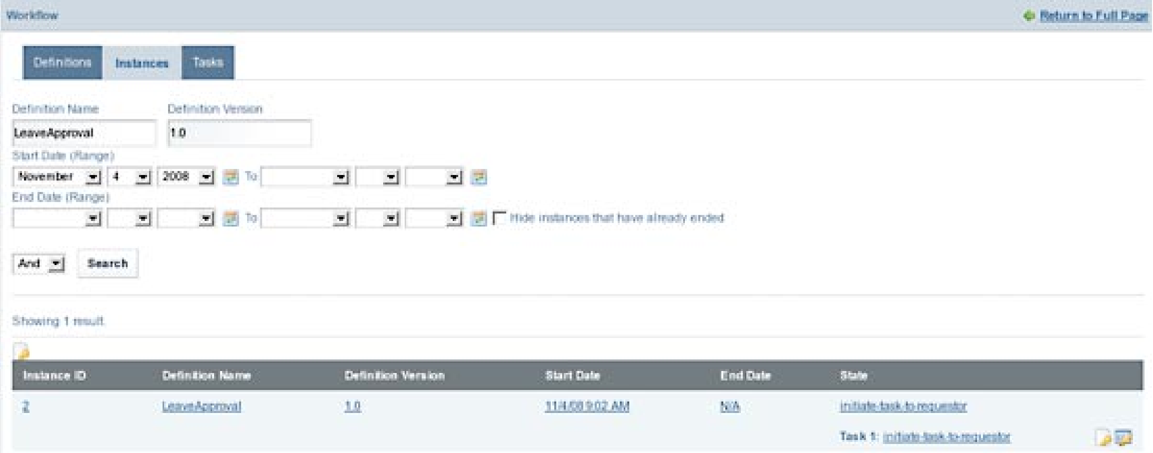


FIGURE 6–31 To Create a Workflow Instance

An instance is created for the Workflow Definition. Provide values for the fields such as the *Definition Name*, *Definition Version*, *Start Date (Range)*, and *End Date (Range)*, and Click *Search* to search for the instance. To hide the instances that have already ended, select the *Hide the instances that have already ended* checkbox.

You can create any number of instances for a version of Workflow Definition.



Workflow

Return to Full Page

Definitions | Instances | Tasks

Definition Name: LeaveApproval

Definition Version: 1.0

Start Date (Range): November 4 2008 To

End Date (Range): To □ Hide instances that have already ended

And | Search

Showing 1 result.

Instance ID	Definition Name	Definition Version	Start Date	End Date	State
2	LeaveApproval	1.0	11/4/08 9:02 AM	N/A	initiate-task-to-requestor Task 1: initiate-task-to-requestor

FIGURE 6–32 To Create a Workflow Instance

3 Click on the Manage button for the instance.

FIGURE 6–33 Manage button

4 Click Save to save the task.

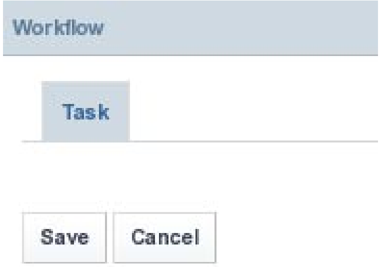


FIGURE 6–34 To Create a Workflow Instance

The task is assigned to the requester.

Note – If the admin user clicks the Signal button against an Instance, the instance is promoted to the next state of workflow. The requester is not allowed to use an instance which is promoted to the next state of the workflow.

Accessing a Workflow Instance

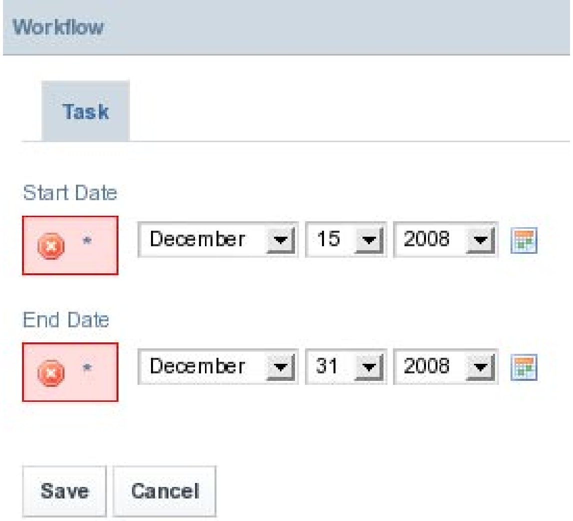
To be done.

▼ Workflow Process Steps: To Request an Holiday

1. **Login to WebSynergy as Paul, the test user.**
2. **Add the Workflow portlet to your page.**
3. **Click on Tasks tab.**

   You can use the *Task Name*, *Definition Name*, *Current Date*, *Start Date*, and *End Date* fields to search for a task. To hide the instances that have already ended, select the *Hide the instances that have already ended* checkbox.
4. **Click on a Task for an Instance which have not ended.**

   In our example, click on the *Holiday Request* task for an Instance. Specify a start date and an end date.



Workflow

Task

Start Date

December 15 2008

End Date

December 31 2008

FIGURE 6–35 Workflow Process Steps: To Request an Holiday

5 **Click Save.**

Workflow Process Steps: To Evaluate Holiday Request

1 **Login to Project WebSynergy as Mary Manager.**

2 **Add the Workflow portlet to your page.**

3 **Click on Tasks tab.**

4 **Click on a Task for an Instance which have not ended.**

In our example, click on the *Evaluate Holiday Request* task for an Instance.

The manager can place Comments against the request. The manager can exercise *Approve*, *Reject*, and *Send Back for Review* options.

The Instance ends on clicking the *Approve* or *Reject* button.

Workflow Process Steps: To Ascertain the Request Status

1 **Login to Project WebSynergy as Paul, the test user.**

2 **Add the Workflow portlet to your page.**

3 **Click on Tasks tab.**

The *End Date* field against the task name has the date and time of approval on it

CHAPTER 7

Sample Use Cases for WebSynergy Portlets

To be done.

Using the Friends Portlet

The following is a use case for adding a friend to the Friends Portlet.

You can view the list of your friends and their profiles on the *Friends* widget. There is no easy and straight forward way of adding friends to the Friends widget.

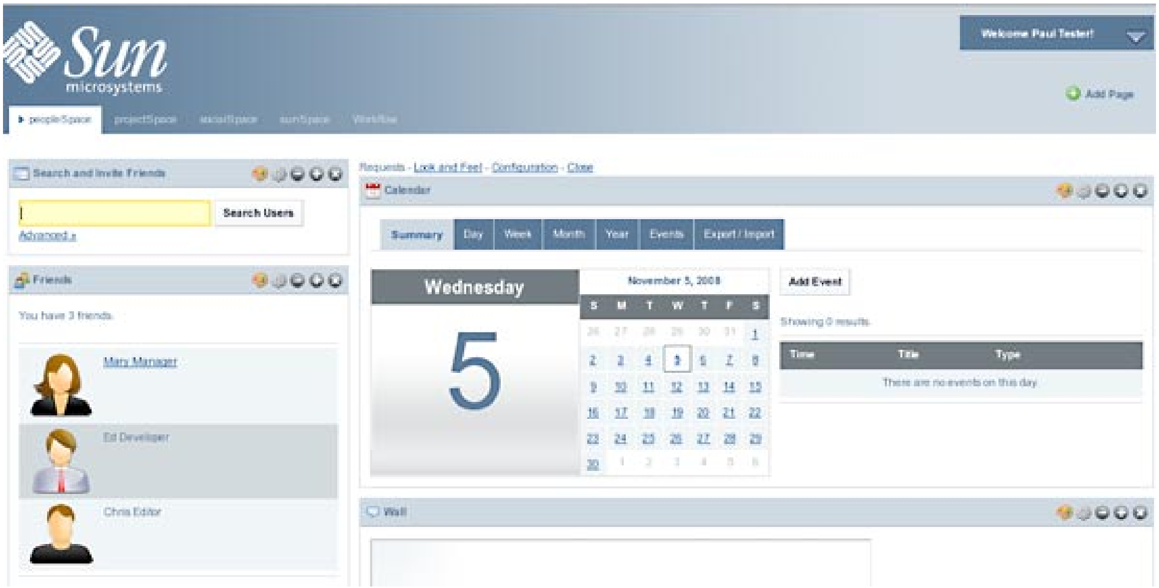
This use case explains a scenario where you make use of the *Search and Invite Friends* widgets and the *Requests* widget to add a friend to the Friends widget.

▼ To Add a Friend

1 **login to WebSynergy as Paul.**

The Friends widget is displaying *Mary manager*, *Ed Developer*, and *Chris Editor* as the friends for *Paul Tester*.

2 **Add the Search and Invite Friends widget to your page.**



You have 3 friends.

November 5, 2008

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

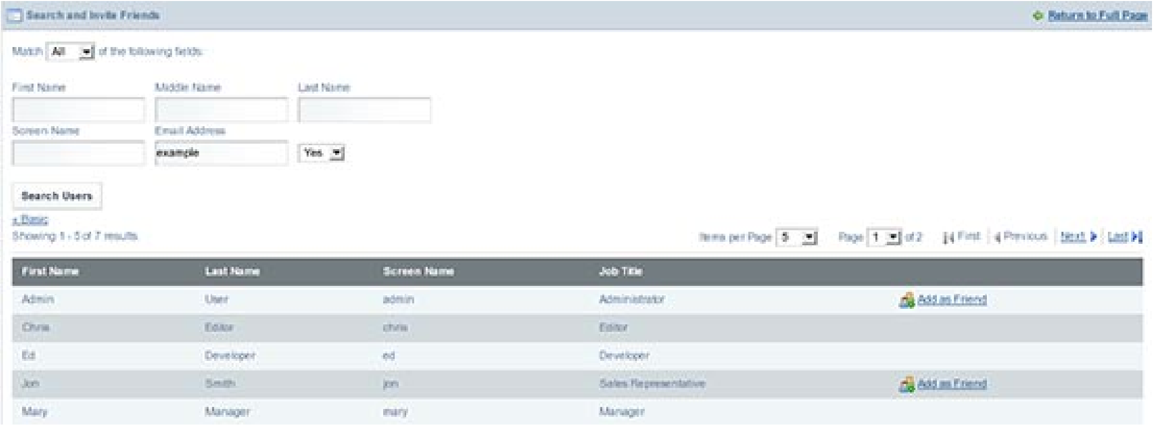
Showing 0 results.

Time	Title	Type
There are no events on this day.		

FIGURE 7–1 To Add a Friend

3 **Click on the Advanced link.**

Specify the search strings which need to be included in you search. For example, type 'example' in the Email Address field to search for users with email id containing the text 'example'.



Search and Invite Friends

Return to Full Page

Match All of the following fields:

First Name	Middle Name	Last Name
Screen Name	Email Address: example	Yes

Search Users

« Basic

Showing 1 - 5 of 7 results.

Items per Page 5 Page 1 of 2 First Previous Next Last

First Name	Last Name	Screen Name	Job Title	
Admin	User	admin	Administrator	Add as Friend
Chris	Editor	chris	Editor	
Ed	Developer	ed	Developer	
Jon	Smith	jon	Sales Representative	Add as Friend
Mary	Manager	mary	Manager	

FIGURE 7–2 To Add a Friend

You can notice that Chris, Ed, and Mary are already friends for Paul, and there is no Add as Friend link against them.

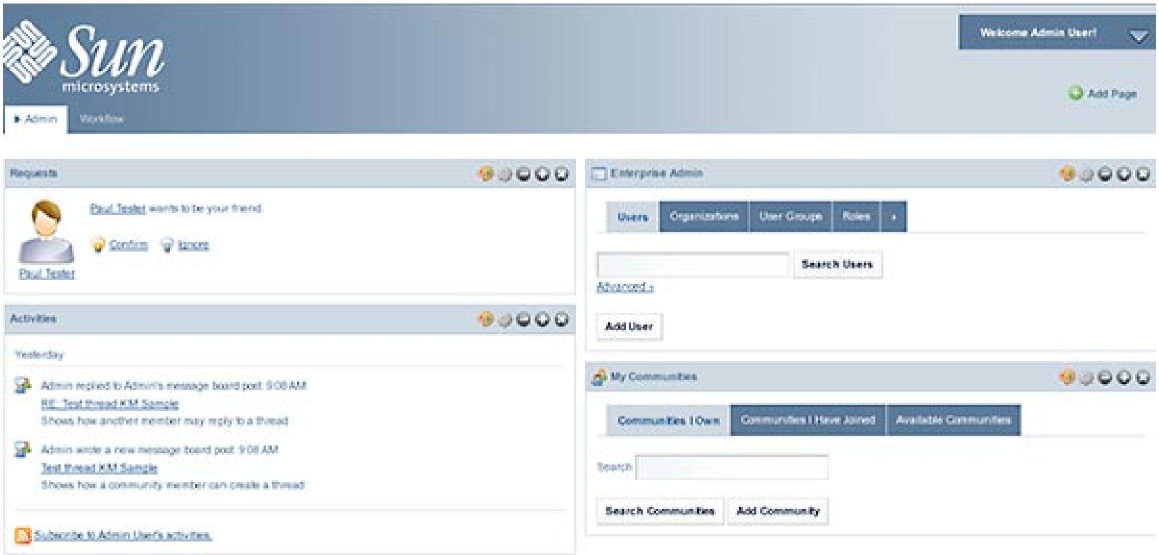
4. **Select a user and click Add as Friend link against the user.**

   In our example, Click *Add as Friend* against the *Admin* user. You get a message saying 'Your request is processed successfully'.

5. **Logout and login again as the user selected as friend.**

   In our example, login as Admin user.

6. **Add the Requests portlet to your page.**



Paul Tester wants to be your friend.

Paul Tester

Yesterday

Admin replied to Admin's message board post. 9:08 AM
RE: Test thread KM Sample
Shows how another member may reply to a thread

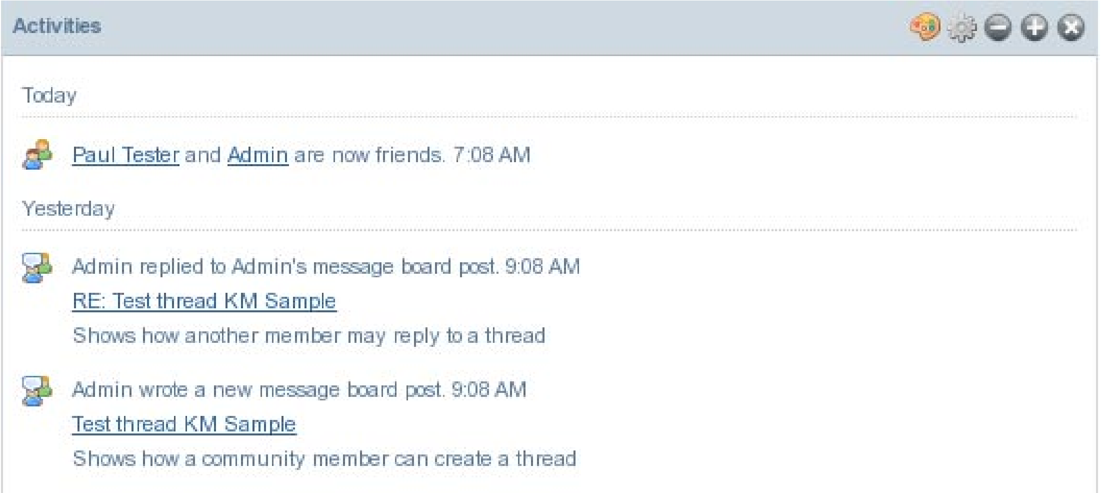
Admin wrote a new message board post. 9:08 AM
Test thread KM Sample
Shows how a community member can create a thread

Subscribe to Admin User's activities.

FIGURE 7–3 To Add a Friend

7. **Click on the Confirm link.**

   You can see on the *Activities* portlet that *Paul Tester and Admin are now friends*.



Today

Paul Tester and Admin are now friends. 7:08 AM

Yesterday

Admin replied to Admin's message board post. 9:08 AM
RE: Test thread KM Sample
Shows how another member may reply to a thread

Admin wrote a new message board post. 9:08 AM
Test thread KM Sample
Shows how a community member can create a thread

FIGURE 7–4 To Add a Friend

The Friends portlet on the Paul Testers page includes Admin as among the friends for Paul.

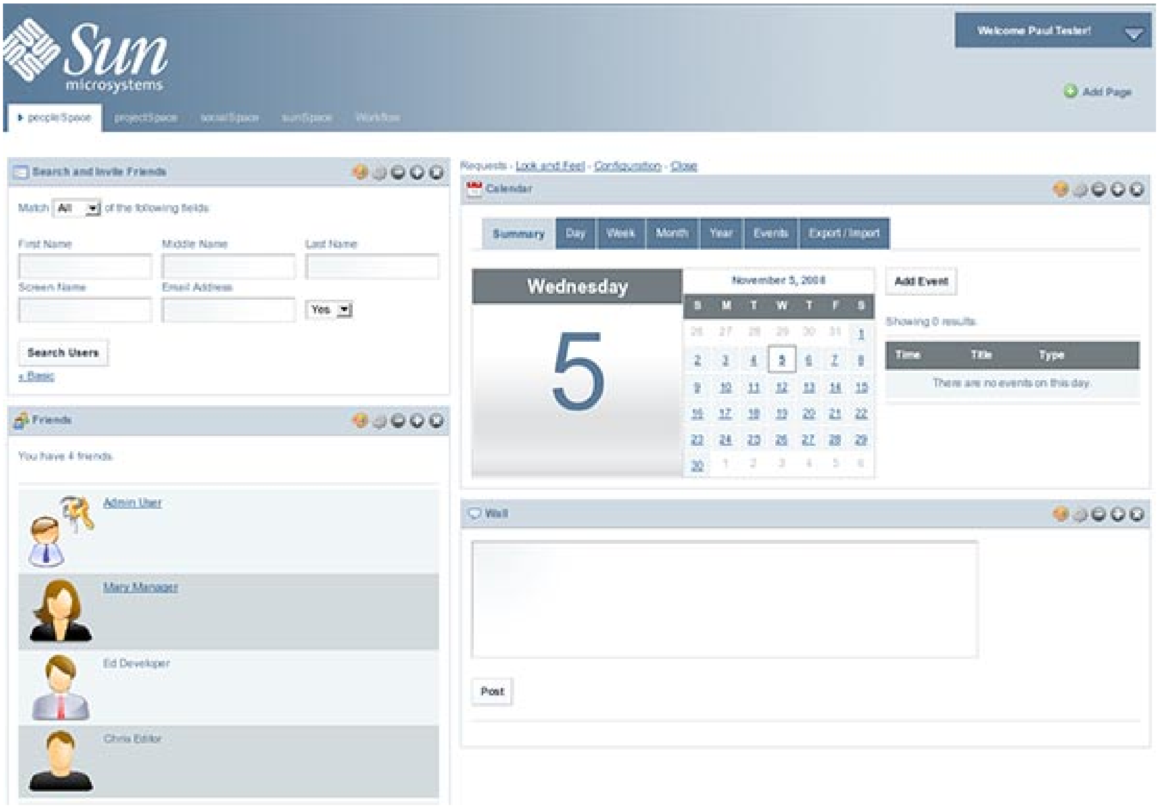


FIGURE 7–5 To Add a Friend

Using the Expense Report Portlet

The Expense Report widget models the Expenses Reports Reclaim process that is typical in any large organization. An Expense Report is a report comprising details of the expenses (like Business Travel, Relocation, Broadband Bills Claim) incurred by an employee that can be reclaimed from the organization.

An Expense Report created by an employee can be approved by his or her superior. Users available to the system created using the *Enterprise Admin Portlet* are mapped to either the Employee role or the Manager role. Let us consider the following Use Case:

Paul Tester (Employee) submits an Expense Report for a business trip. Mary Manager (Manager) verifies and approves the Expense Report. The following is the process involved:

1. Paul Tester submits an Expense Report. Paul performs the following steps to submit his report:

  a. Paul selects the expense category, and enters an expense amount and description and saves the report. The system assigns a unique expense ID to the report.

  b. Paul reviews the report summary.

  c. Paul submits the saved report for approval by his manager.

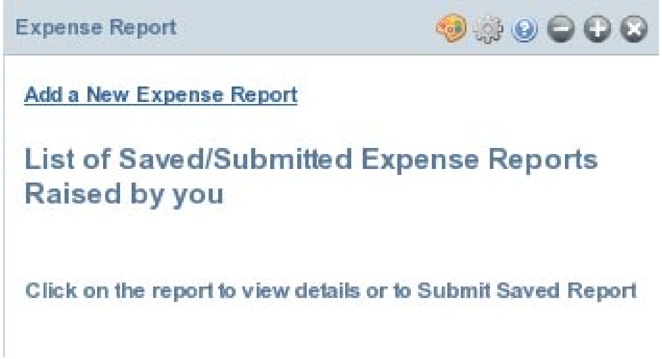
2. Mary Manager verifies the details of the Expense Report. She can either approve or reject the report.

- "To Submit an Expense Report" on page 230
- "To Approve an Expense Report" on page 234

▼ To Submit an Expense Report

1 Login to WebSynergy as Paul Tester.

2 Add the Expense Report Portlet to your page.



Expense Report

Add a New Expense Report

List of Saved/Submitted Expense Reports Raised by you

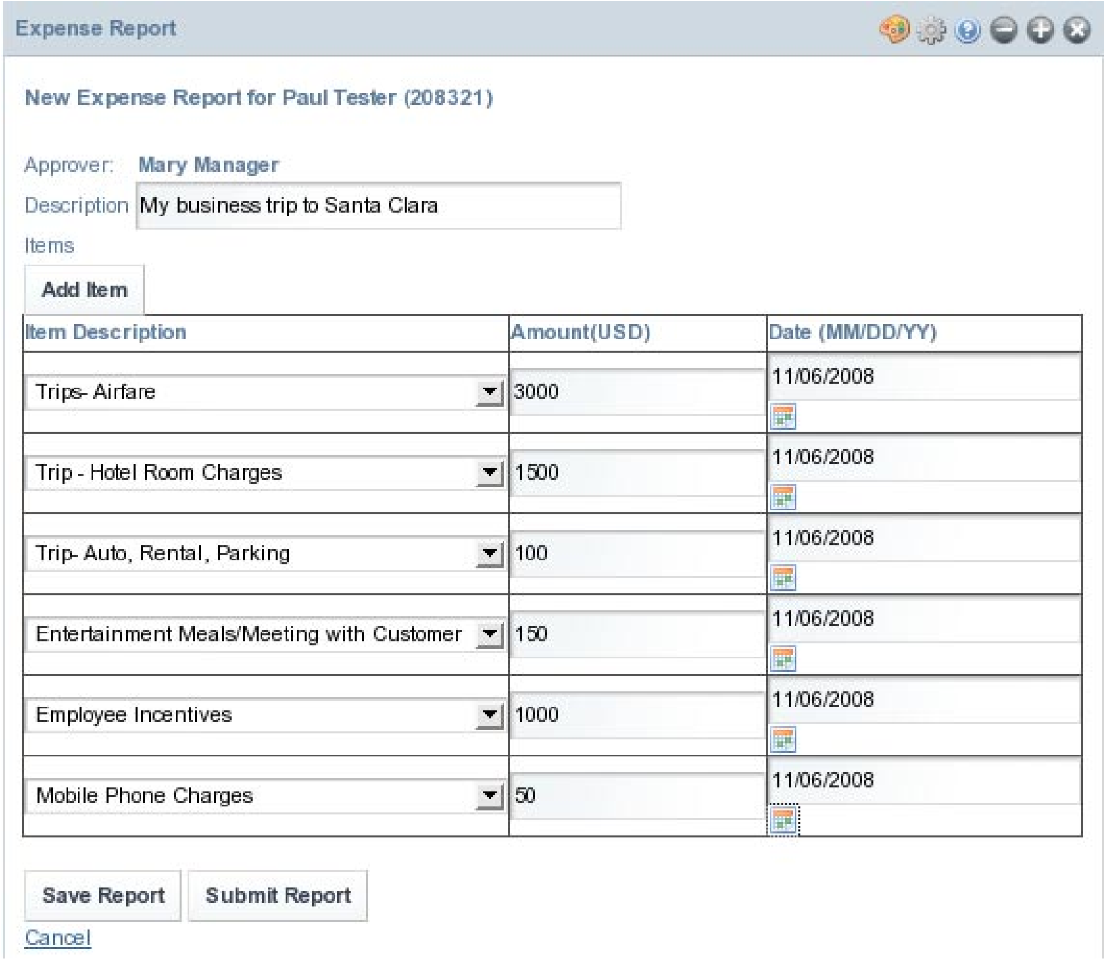
Click on the report to view details or to Submit Saved Report

FIGURE 7–6 To Submit an Expense Report

3 Click on the Add a New Expense Report link.

4 Enter a description for the Expense Report.

Enter values for *Item Description*, *Amount*, and *Date* fields. Click on *Add Item* to add a new row.



Expense Report

New Expense Report for Paul Tester (208321)

Approver: Mary Manager

Description: My business trip to Santa Clara

Items

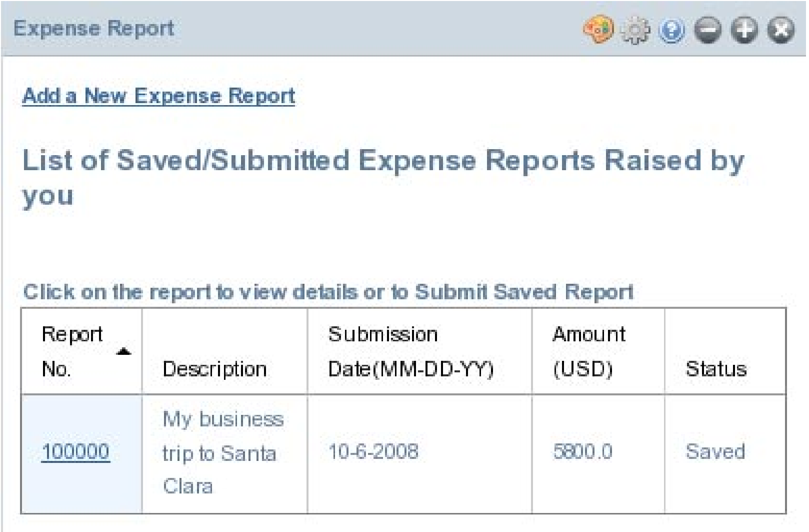
Item Description	Amount(USD)	Date (MM/DD/YY)
Trips- Airfare	3000	11/06/2008
Trip - Hotel Room Charges	1500	11/06/2008
Trip- Auto, Rental, Parking	100	11/06/2008
Entertainment Meals/Meeting with Customer	150	11/06/2008
Employee Incentives	1000	11/06/2008
Mobile Phone Charges	50	11/06/2008

FIGURE 7–7 To Submit an Expense Report

5 **Click on Save Report.**

6 **Click on the Go back to Report Summary link to view the report summary.**

The summary page displays the *Report No.*, *Description*, *Submission Date*, *Amount*, and the *Status*.



Expense Report

Add a New Expense Report

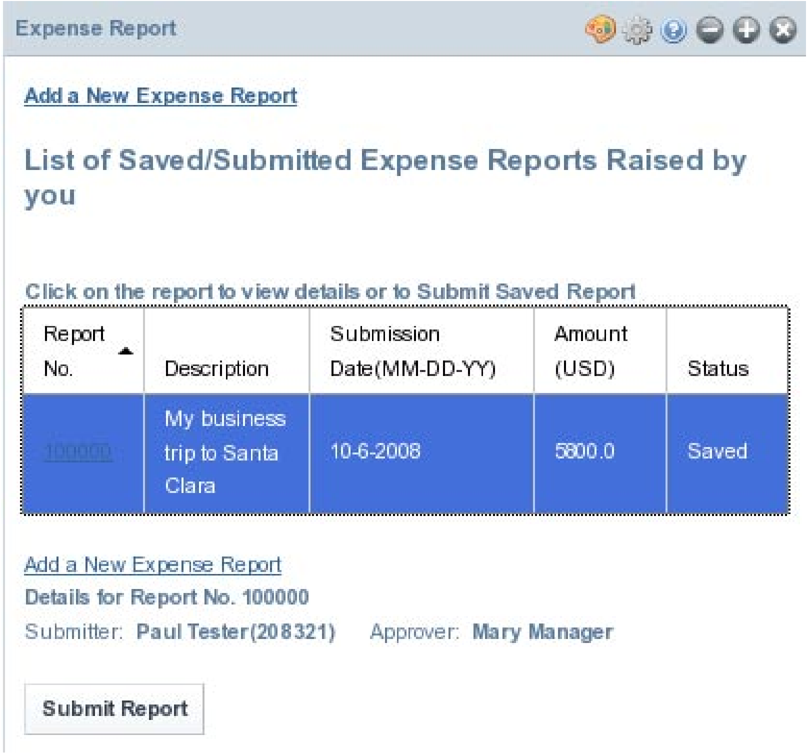
List of Saved/Submitted Expense Reports Raised by you

Click on the report to view details or to Submit Saved Report

Report No. ▲	Description	Submission Date(MM-DD-YY)	Amount (USD)	Status
100000	My business trip to Santa Clara	10-6-2008	5800.0	Saved

FIGURE 7–8 To Submit an Expense Report

Click on the report ID. The report is enabled for submit.



Add a New Expense Report

List of Saved/Submitted Expense Reports Raised by you

Click on the report to view details or to Submit Saved Report

Report No.	Description	Submission Date(MM-DD-YY)	Amount (USD)	Status
100000	My business trip to Santa Clara	10-6-2008	5800.0	Saved

Add a New Expense Report

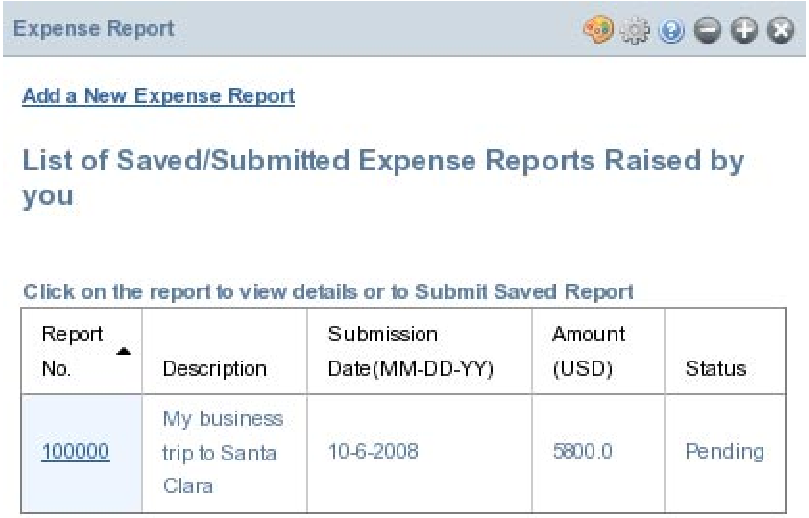
Details for Report No. 100000

Submitter: Paul Tester(208321) Approver: Mary Manager

FIGURE 7–9 To Submit an Expense Report

7 **Click Submit Report.**

The report is submitted for approval. Status of the report changes to *Pending*.



Expense Report

Add a New Expense Report

List of Saved/Submitted Expense Reports Raised by you

Click on the report to view details or to Submit Saved Report

Report No.	Description	Submission Date(MM-DD-YY)	Amount (USD)	Status
100000	My business trip to Santa Clara	10-6-2008	5800.0	Pending

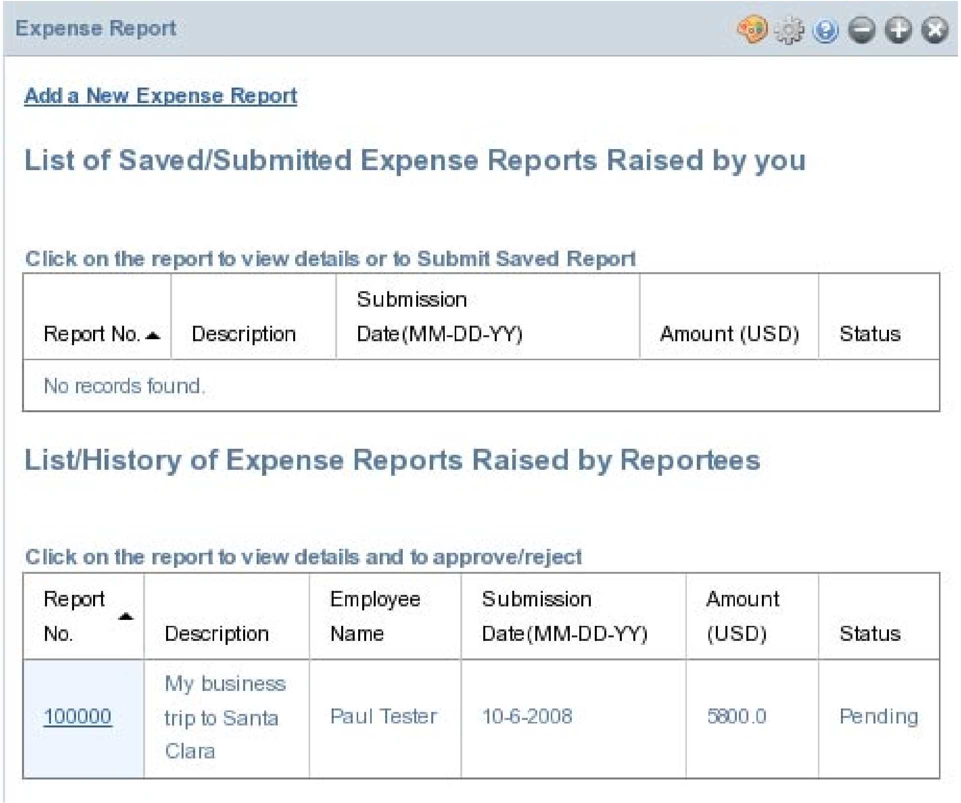
FIGURE 7–10 To Submit an Expense Report

▼ To Approve an Expense Report

1. **Login to WebSynergy as Mary Manager.**

2. **Add the Expense Report Portlet to your page.**

   The report submitted by Paul is displayed under the *List/History of Expense Reports Raised by Reportees*.



Expense Report

Add a New Expense Report

List of Saved/Submitted Expense Reports Raised by you

Click on the report to view details or to Submit Saved Report

Report No. ▲	Description	Submission Date(MM-DD-YY)	Amount (USD)	Status
No records found.				

List/History of Expense Reports Raised by Reportees

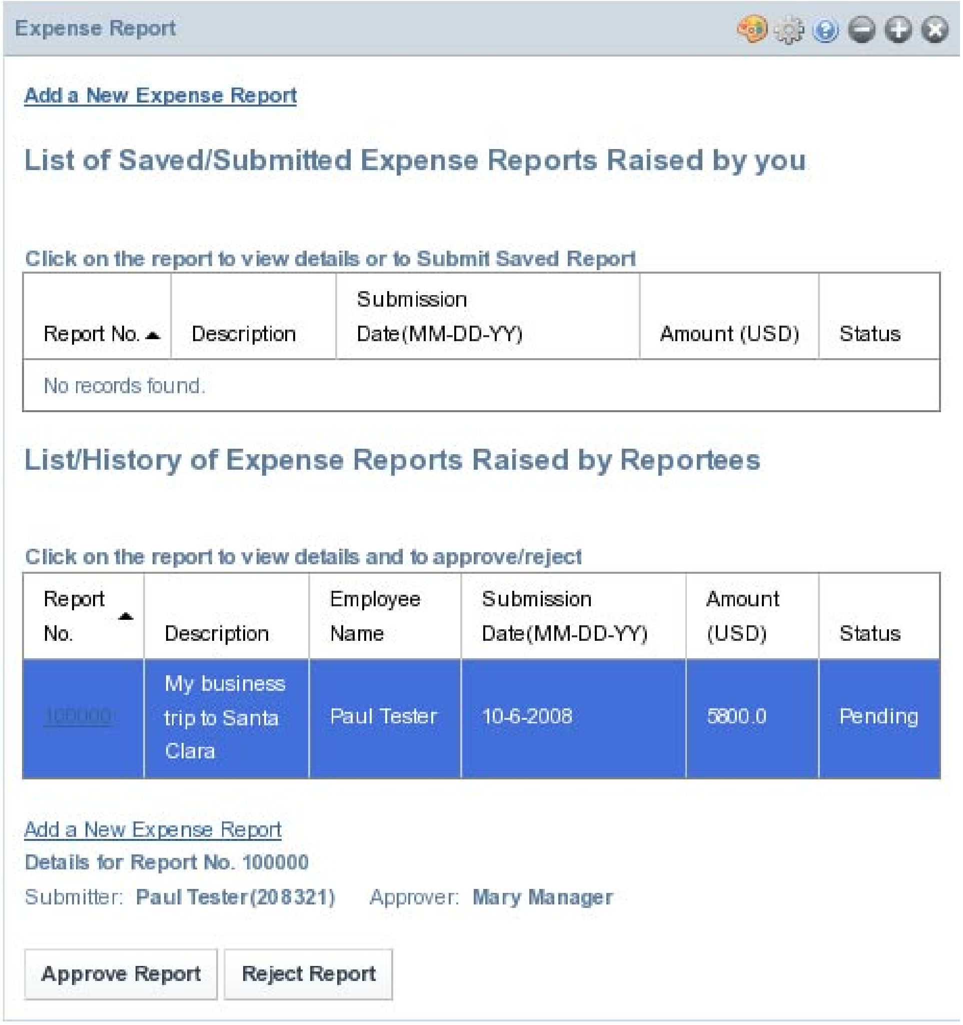
Click on the report to view details and to approve/reject

Report No. ▲	Description	Employee Name	Submission Date(MM-DD-YY)	Amount (USD)	Status
100000	My business trip to Santa Clara	Paul Tester	10-6-2008	5800.0	Pending

FIGURE 7–11 To Approve an Expense Report

3 **Click on the report ID.**

Details of the Submitter are displayed. Also, the *Approve Report* and *Reject Report* buttons appear.



Add a New Expense Report

List of Saved/Submitted Expense Reports Raised by you

Click on the report to view details or to Submit Saved Report

Report No. ▲	Description	Submission Date(MM-DD-YY)	Amount (USD)	Status
No records found.				

List/History of Expense Reports Raised by Reportees

Click on the report to view details and to approve/reject

Report No. ▲	Description	Employee Name	Submission Date(MM-DD-YY)	Amount (USD)	Status
100000	My business trip to Santa Clara	Paul Tester	10-6-2008	5800.0	Pending

Add a New Expense Report

Details for Report No. 100000

Submitter: **Paul Tester(208321)** Approver: **Mary Manager**

FIGURE 7–12 To Approve an Expense Report

4 **Click on the Approve Report button to approve the report.**

Delegating Portlets to Users

A User has a number of Roles associated to him. Each portlet is also associated with one or more Roles. A user can access only the portlets which are associated with his roles. If the administrator (Admin User) assigns a new role to a user, all the portlets which are associated with the role are accessible by the user. If the administrator creates a new role and associates it with some portlets, those portlets will be accessible by any of the users who have this new role assigned to them. Like this, roles can be used to delegate portlets to users.

A portlet has one or more of the four roles (Guest, Power User, User, and Administrator) by default. You can associate more roles to a portlet. For example, you can add 'Power User' and 'User' roles to a portlet which has 'Guest' role associated with it. In this case, any user with either 'Guest', 'Power User', or 'User' role will be able to access the portlet.

Consider the following scenario:

The user 'Chris Editor' has 'Power User' and 'User' Roles assigned to him. He can access only the portlets with Power User and User roles. The 'Admin portlet' has only the 'Administrator' role assigned to it, and only the 'Admin User' can access that portlet. Creating a new role named 'DelegatedAdmin' and assigning it to the Admin portlet. Then, assign the DelegatedAdmin role to Chris Editor. Now, Chris Editor has Power User, User, and DelegatedAdmin roles. Chris Editor becomes the delegated administrator for the Admin Portlet. Chris Editor can access the Admin Portlet.

This following are the steps involved:

1. Login to WebSynergy as Admin User and use the Enterprise Admin portlet to create a new role named 'DelegatedAdmin'.
2. Add the DelegatedAdmin role to the Admin Portlet.
3. Assign the DelegatedAdmin role to Chris Editor.
4. Login to WebSynergy as Chris Editor. You can access the Admin Portlet.

The following are the tasks involved:

1. “To Create a new Role” on page 237
2. “To Assign a new Role to a Portlet” on page 238
3. “To Assign a new Role to a User” on page 241
4. “To Access a Portlet” on page 243

▼ To Create a new Role

1 Login to WebSynergy as Admin User.

2 Click on the Roles tab in the Enterprise Admin portlet.

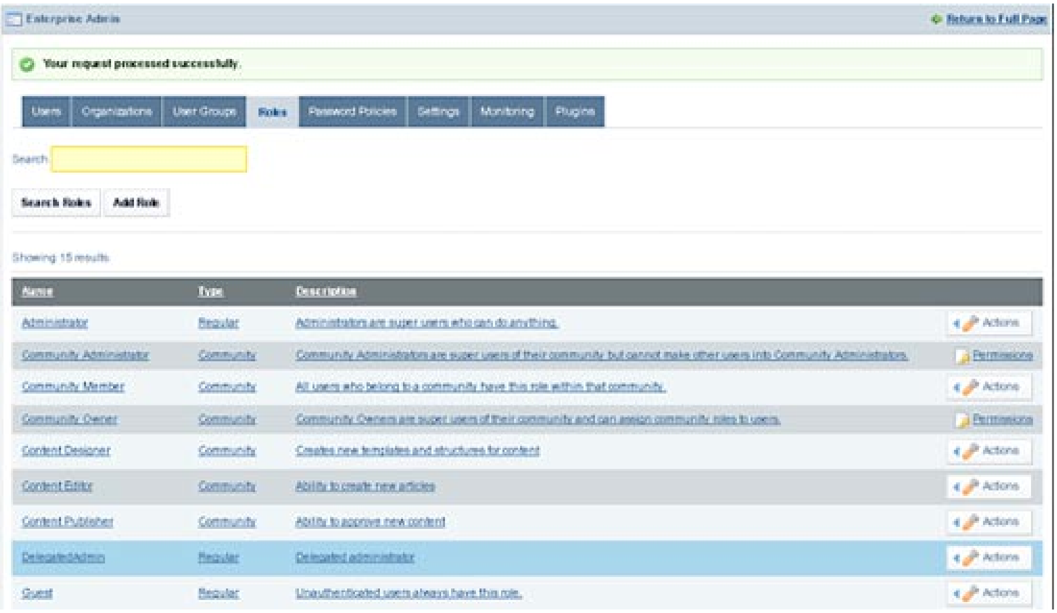
3 **Click on the Add Role button.**

4 **Specify a name for the role.**

In this example, DelegatedAdmin is the name of the role.

5 **Click Save.**

The new role is added to the list of roles.



Your request processed successfully.

Showing 15 results.

Name	Type	Description	
Administrator	Regular	Administrators are super users who can do anything.	Actions
Community Administrator	Community	Community Administrators are super users of their community but cannot make other users into Community Administrators.	Permissions
Community Member	Community	All users who belong to a community have this role within that community.	Actions
Community Owner	Community	Community Owners are super users of their community and can assign community roles to users.	Permissions
Content Designer	Community	Creates new templates and structures for content	Actions
Content Editor	Community	Ability to create new articles	Actions
Content Publisher	Community	Ability to approve new content	Actions
DelegatedAdmin	Regular	Delegated administrator	Actions
Guest	Regular	Unauthenticated users always have this role.	Actions

FIGURE 7–13 To Create a new Role

To Assign a new Role to a Portlet

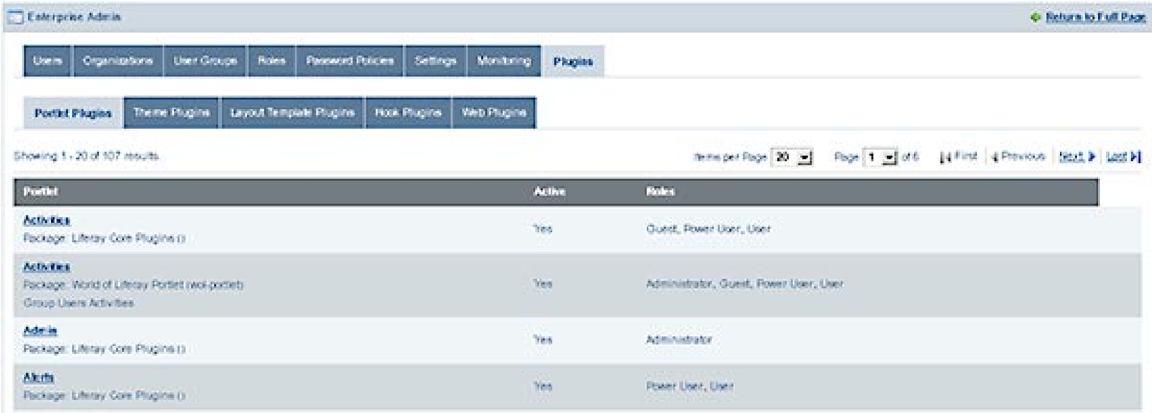
1 **Login to WebSynergy as Admin User.**

2 **click any of the tabs in the Enterprise Admin portlet.**

The Enterprise Admin portlet expands.

3 **Click on the plugins tab.**

All plugins are listed. The first tab displays portlet plugins.



Showing 1 - 20 of 107 results.

Items per Page 20 Page 1 of 6 First Previous Next Last

Portlet	Active	Roles
Activities Package: Liferay Core Plugins ()	Yes	Guest, Power User, User
Activities Package: World of Liferay Portlet (wol-portlet) Group Users Activities	Yes	Administrator, Guest, Power User, User
Admin Package: Liferay Core Plugins ()	Yes	Administrator
Alerts Package: Liferay Core Plugins ()	Yes	Power User, User

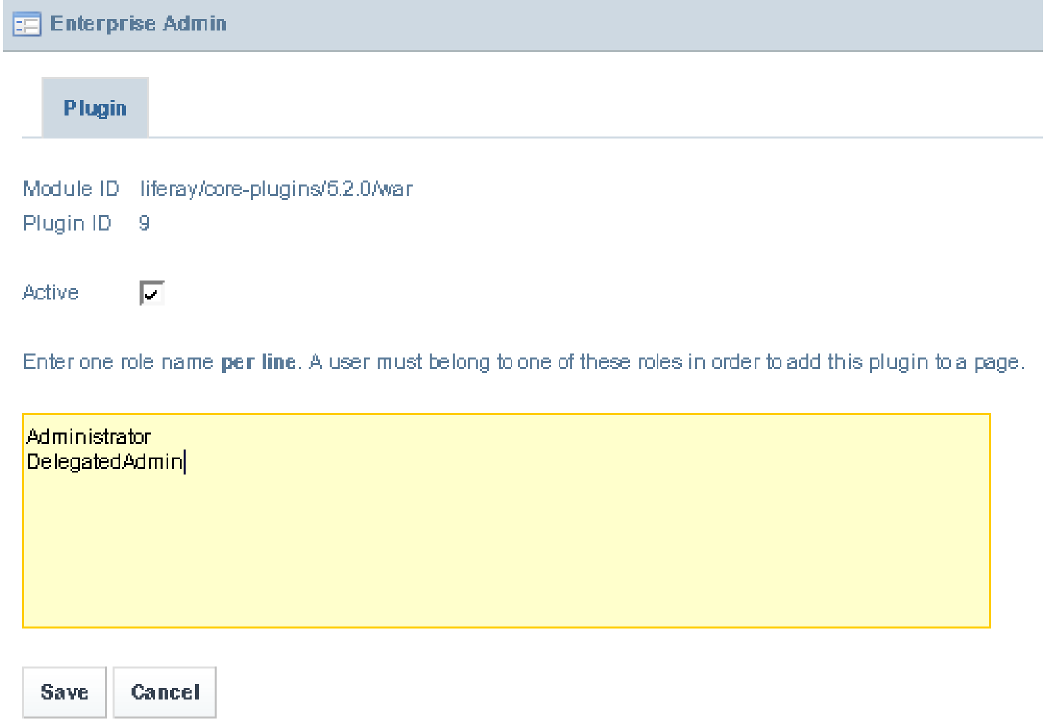
FIGURE 7–14 To Create a new Role

4 **Click on a portlet plugin.**

In this example, click on 'Admin'. The edit page for plugin attributes appears.

5 **Add a new role for the plugin.**

In this example, add the 'DelegatedAdmin' role.



Module ID liferay/core-plugins/5.2.0/war
Plugin ID 9

Active ☑

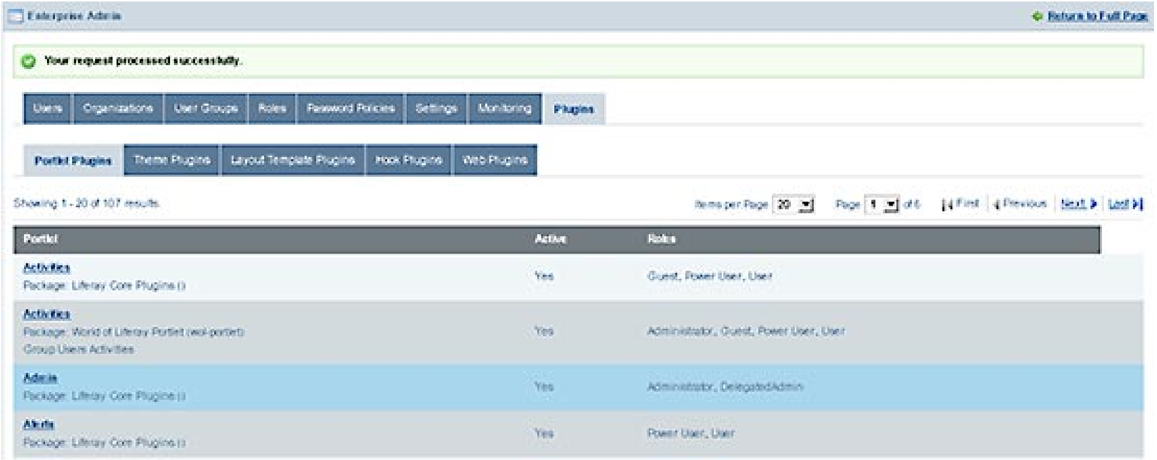
Enter one role name **per line**. A user must belong to one of these roles in order to add this plugin to a page.

```
Administrator
DelegatedAdmin
```

FIGURE 7–15 To Create a new Role

6. **Click Save.**

   The new role is added to the portlet plugin.



Your request processed successfully.

Showing 1 - 20 of 107 results.

Portlet	Active	Roles
Activities Package: Liferay Core Plugins ()	Yes	Guest, Power User, User
Activities Package: World of Liferay Portlet (wol-portlet) Group Users Activities	Yes	Administrator, Guest, Power User, User
Admin Package: Liferay Core Plugins ()	Yes	Administrator, DelegatedAdmin
Alerts Package: Liferay Core Plugins ()	Yes	Power User, User

FIGURE 7–16 To Create a new Role

▼ To Assign a new Role to a User

1 **Login to WebSynergy as Admin User.**

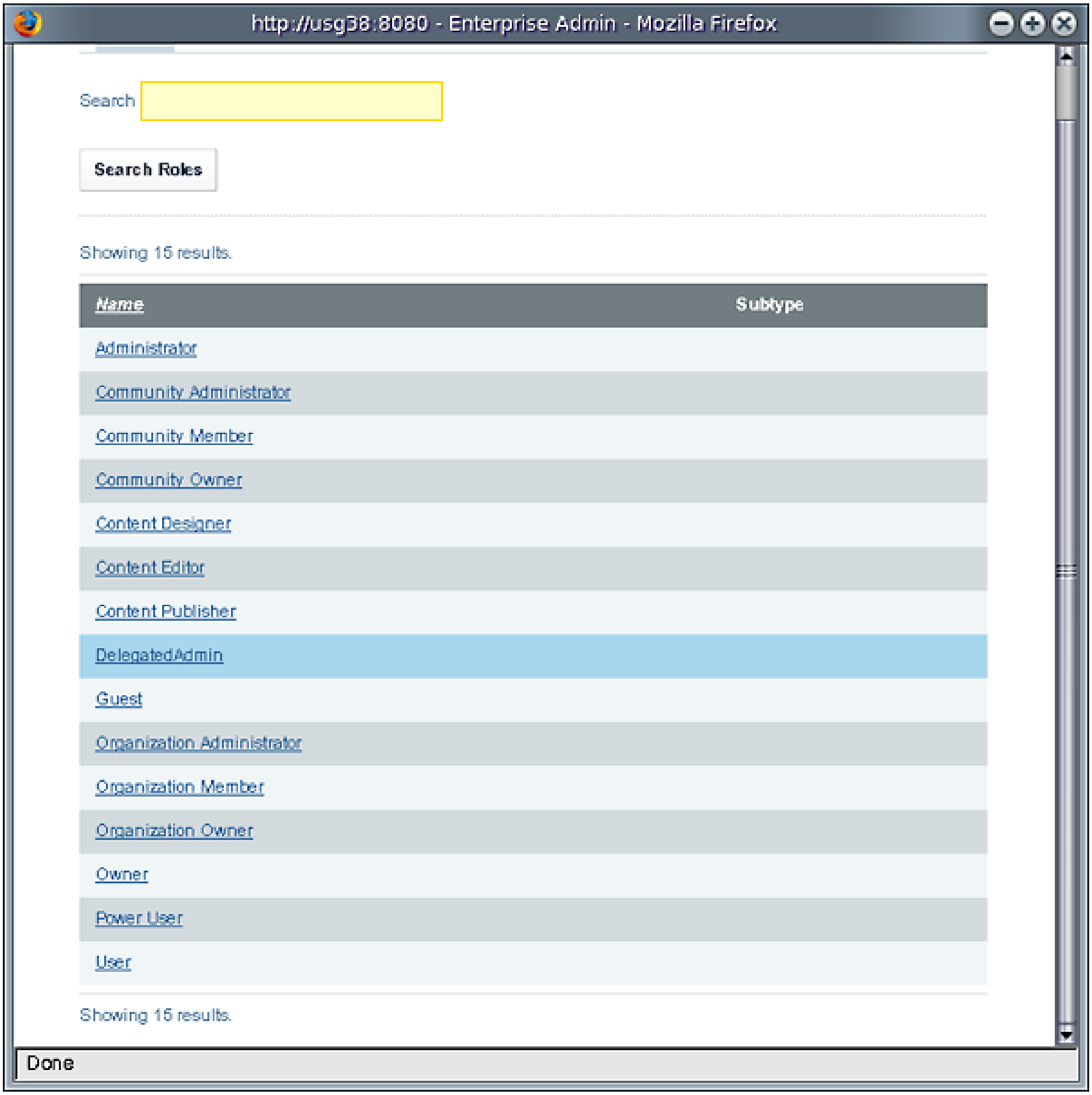
2 **Click on the Users tab.**

3 **Click on a user.**

In this example, choose 'Chris'. The page for the user appears

4 **Click on the Select button below Roles.**

The window listing all available roles appears.



Showing 15 results.

Name	Subtype
Administrator	
Community Administrator	
Community Member	
Community Owner	
Content Designer	
Content Editor	
Content Publisher	
DelegatedAdmin	
Guest	
Organization Administrator	
Organization Member	
Organization Owner	
Owner	
Power User	
User	

Showing 15 results.

FIGURE 7–17 To Assign a new Role to a User

5. **Click on a selected Role.**

   In this example, click on 'DelegatedAdmin'. The Role is assigned to the User.

6. **Click Save.**

▼ To Access a Portlet

1 **Login to WebSynergy as a general user.**

In this example, login as Chris Editor.

2 **Click Add Application form the welcome menu.**

3 **Search for the delegated portlet using the search box in the Add Application window.**

In this example, search for 'Admin'.

4 **Add the portlet to your page by clicking on the Add button against the portlet.**

In this example, add the Admin portlet to your page.

Using the Presence Feature

To be done.

- "Chatting Using Project WebSynergy" on page 243
- "Making Calls Using Project WebSynergy" on page 244

Chatting Using Project WebSynergy

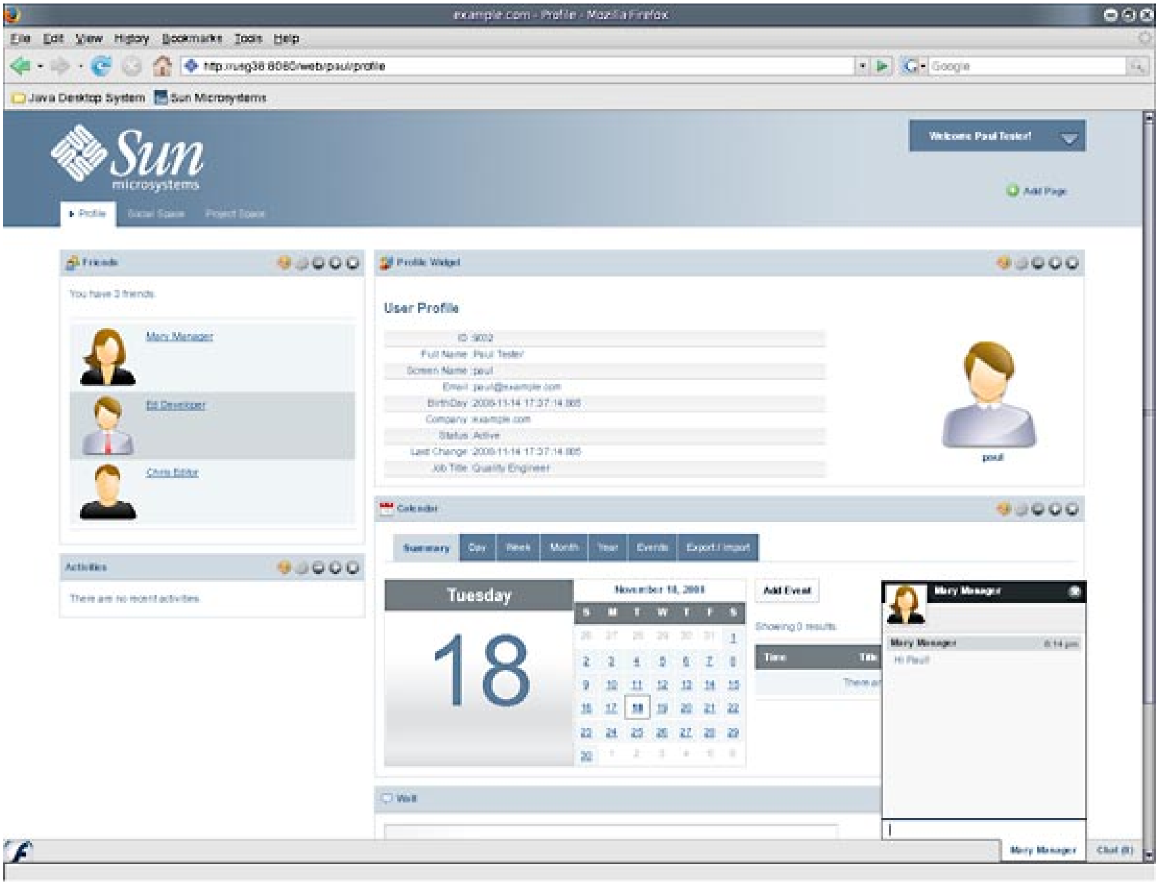
The 'Chat' feature in Project WebSynergy enables you to chat with the 'Friends' who are currently online.

If you consider the example of *Sample Users* for Project WebSynergy, *Ed Developer*, *Mary Manager*, and *Chris Editor* are the Friends for *Paul Tester*. The Friends portlet displays all the friends for the user.

This means, Paul can chat with Ed, Mary or Chris if they are online. You can chat with friends by clicking on *Chat* at the right-bottom of your page. *Chat {0}* indicates that no friends are online. *Chat {1}* indicates that there is one friend online.

Consider, a simple use case to demonstrate the working on the Chat feature.

1. Open a browser window and login to Project WebSynergy as Paul Tester.
2. Open another browser window and login as Mary Manager.
3. Click on *Chat*. It displays all the friends who are online. In this example, it displays 'Paul Tester'. Click on 'Paul Tester' to write a message, and press Enter.
4. When you login again as Paul Tester, you can view the message form Mary Manager by clicking on Chat. You can reply to the message from Mary Manager.



You have 3 friends.

Mary Manager

Ed Developer

Chris Editor

There are no recent activities.

User Profile

ID: 9002
Full Name: Paul Tester
Screen Name: paul
Email: paul@example.com
BirthDay: 2008-11-14 17:37:14.805
Company: example.com
Status: Active
Last Change: 2008-11-14 17:37:14.805
Job Title: Quality Engineer

paul

Tuesday

18

November 18, 2008

S	M	T	W	T	F	S
26	27	28	29	30	31	1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29
30	1	2	3	4	5	6

Showing 0 results.

There ar

Mary Manager 6:14 pm
Hi Paul!

FIGURE 7–18 Chat feature

Making Calls Using Project WebSynergy

To be done.

Using the Portlets for Enterprise Social Networking

Facebook, *Linkedin*, *Plaxo*, and *Twitter* are the most popular Enterprise Social Networking websites. The Facebook, Linkedin, Plaxo, and Twitter portlets inWebSynergy provide the interface to work with these Enterprise Social Networking websites respectively.

- "To Access the Facebook Portlet" on page 245
- "To Access the Linkedin Portlet" on page 245

Similarly, you can access Plaxo and Twitter.

▼ To Access the Facebook Portlet

1 **Login to WebSynergy as a general user.**

In this example, login as Chris Editor.

2 **Click Add Application form the welcome menu.**

3 **Add the Facebook portlet to your page.**

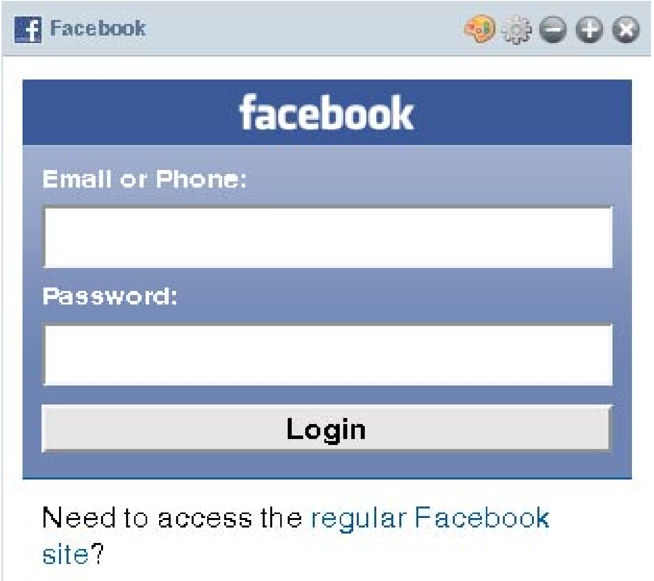


FIGURE 7–19 Facebook portlet

4 **Login to Facebook by using the interface provided by the portlet.**

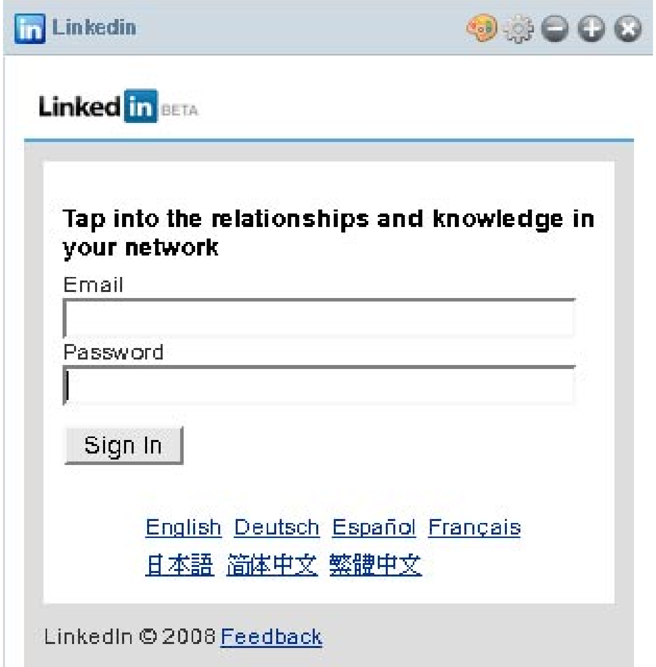
▼ To Access the Linkedin Portlet

1 **Login to WebSynergy as a general user.**

In this example, login as Chris Editor.

2 **Click Add Application form the welcome menu.**

3 **Add the Linkedin portlet to your page.**



Linkedin

LinkedIn BETA

Tap into the relationships and knowledge in your network

Email

Password

Sign In

English Deutsch Español Français

日本語 简体中文 繁體中文

LinkedIn © 2008 Feedback

FIGURE 7–20 Linkedin portlet

4 **Login to Facebook by using the interface provided by the portlet.**

Working with Communities and Organizations

To be done.

- "Adding Public and Private Pages" on page 246
- "Activating Staging and Workflow" on page 249
- "Enabling Crawling of Page URLs by Search Engines" on page 252

Adding Public and Private Pages

You can add public and private pages to Communities and Organizations.

The below procedure explains how you can add a public page and a private page to an Organization. You can add public page and private pages to a Community by following the similar procedure.

For the procedure to create an Organization, see "To Create an Organization" on page 74.

▼ To add a Public and a Private Page to an Organization

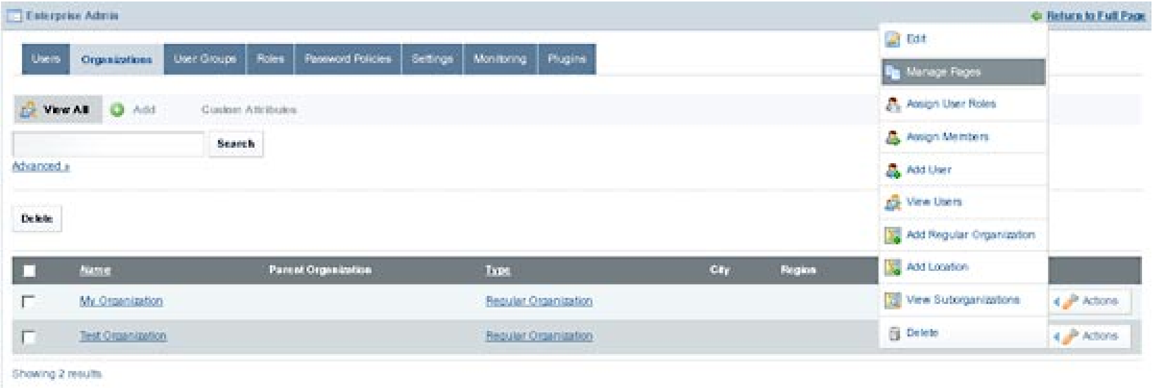
1. **Login to Project WebSynergy as Admin User.**

2. **Click on the Organizations tab in the Enterprise Admin portlet.**

   All existing Organizations are listed.

3. **To add a page, click on the Actions button against an Organization and choose Manage Pages from the menu.**

   In this example, we are adding pages to an Organization called 'Test Organization'.



□	Name	Parent Organization	Type	City	Region	
□	My Organization		Regular Organization			Actions
□	Test Organization		Regular Organization			Actions

Showing 2 results

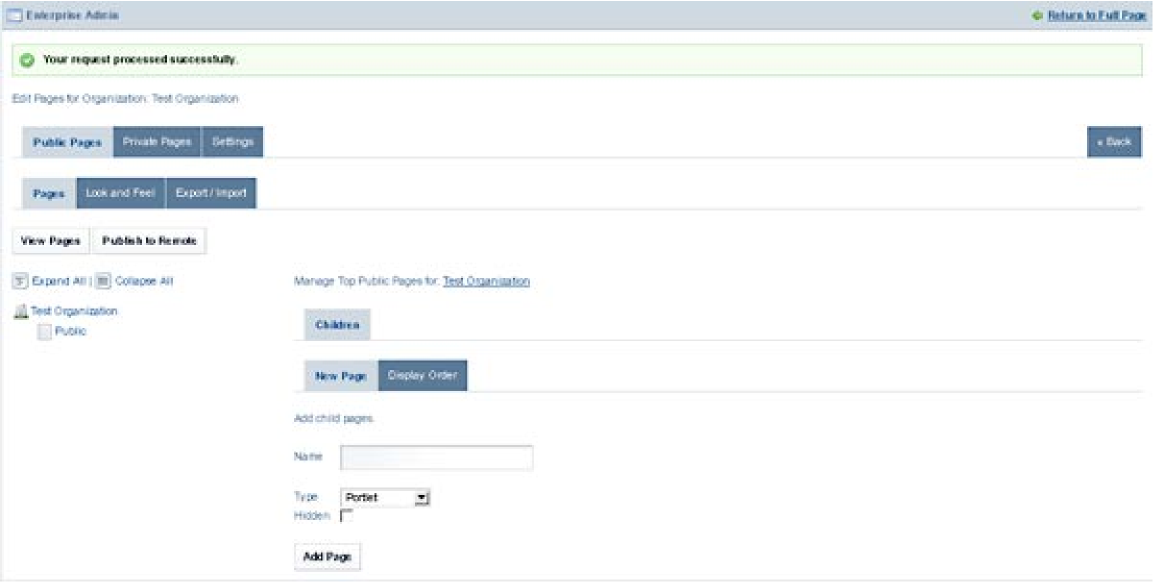
FIGURE 7–21 Adding a page

4. **To add a public page, choose the Public Pages tab and specify Name and Type for the page.**

   In this example, we are creating a public page named as 'Public' with the type 'portlet'

5. **Click on the Add Page button.**

   The page is added to the Organization.



Your request processed successfully.

Edit Pages for Organization: Test Organization

Manage Top Public Pages for: Test Organization

Add child pages.

Name

Type: Portlet

Hidden □

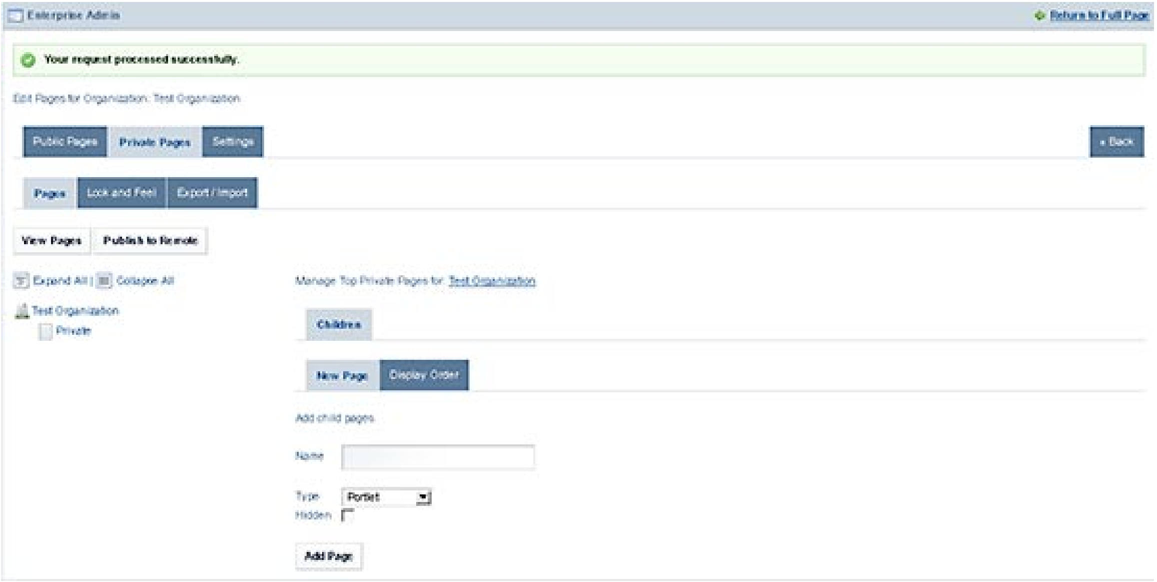
FIGURE 7–22 Adding a page

6 **To add a private page, click on the Private Pages tab and specify Name and Type for the page.**

In this example, we are creating a private page named as 'Private' with the type 'portlet'

7 **Click on the Add Page button.**

The page is added to the Organization.



Your request processed successfully.

Edit Pages for Organization: Test Organization

Manage Top Private Pages for: Test Organization

Add child pages

Name

Type Portlet

Hidden □

FIGURE 7–23 Adding a page

You can add more Public or Private pages and can change the display order for pages by clicking on the *Display Order* tab.

Activating Staging and Workflow

To be done.

Activating Staging

The admin user can activate staging for *Communities* and *Organizations*. When you activate staging for Communities or Organizations, they can be used in the production environment.

WebSynergy by default has *Content Management Sample*, *Knowledge Management Sample*, and *Collaboration Sample* Communities. For the procedure to create a new Community, see .

The below procedure explains how we can stage Communities. You can stage Organizations by following the similar procedure.

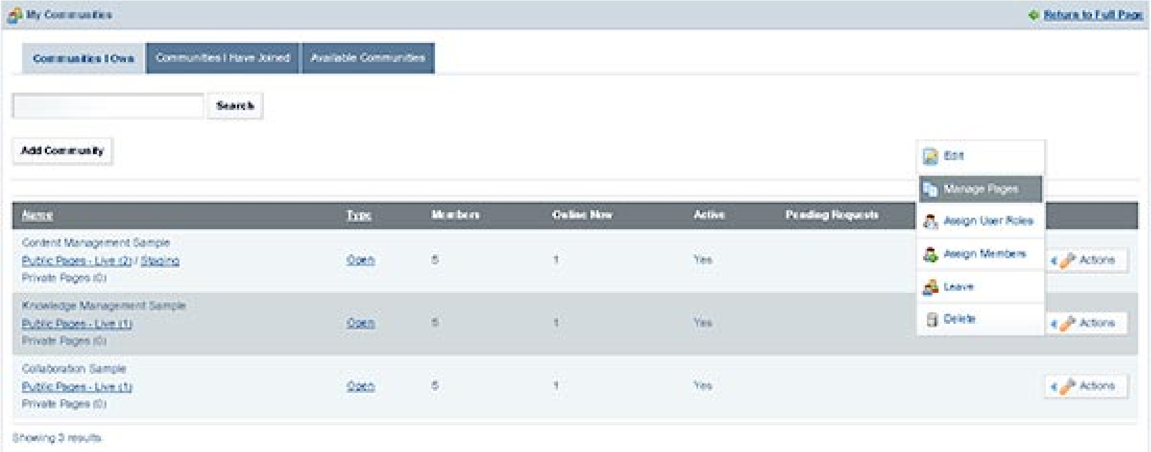
▼ To Stage a Community

1. **Login to Project WebSynergy as Admin User.**

2. **Click on the Communities I Own tab in the My Communities portlet.**

3. **To stage a Community, click on the Actions button against a Community and choose Manage Pages from the menu.**

   In this example, choose the 'Knowledge Management Sample' Community.



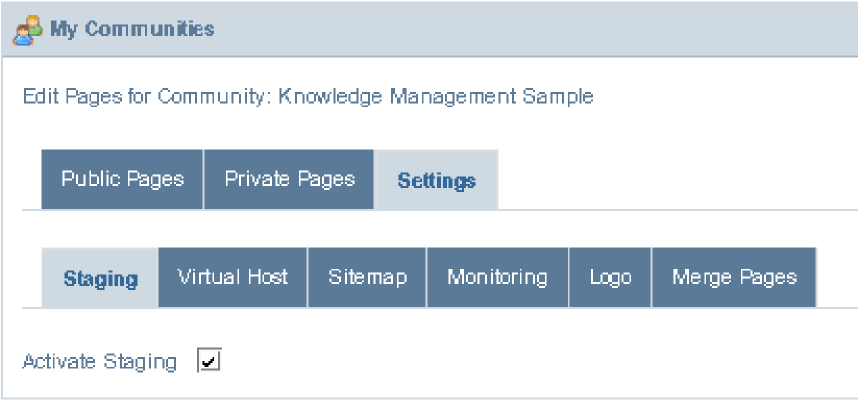
Name	Type	Members	Online Now	Active	Pending Requests	
Content Management Sample Public Pages - Live (2) / Staging Private Pages (0)	Open	5	1	Yes		Actions
Knowledge Management Sample Public Pages - Live (1) Private Pages (0)	Open	5	1	Yes		Actions
Collaboration Sample Public Pages - Live (1) Private Pages (0)	Open	5	1	Yes		Actions

Showing 3 results.

FIGURE 7–24 Staging Communities

4. **Click on the Settings tab, and enable the Activate Staging option.**

   The community is staged to the production environment.



Edit Pages for Community: Knowledge Management Sample

Activate Staging ☑

FIGURE 7–25 Staging Communities

5. **Choose My Places from the welcome menu and navigate to the page.**

   The live page for 'Knowledge Management Sample' is displayed.

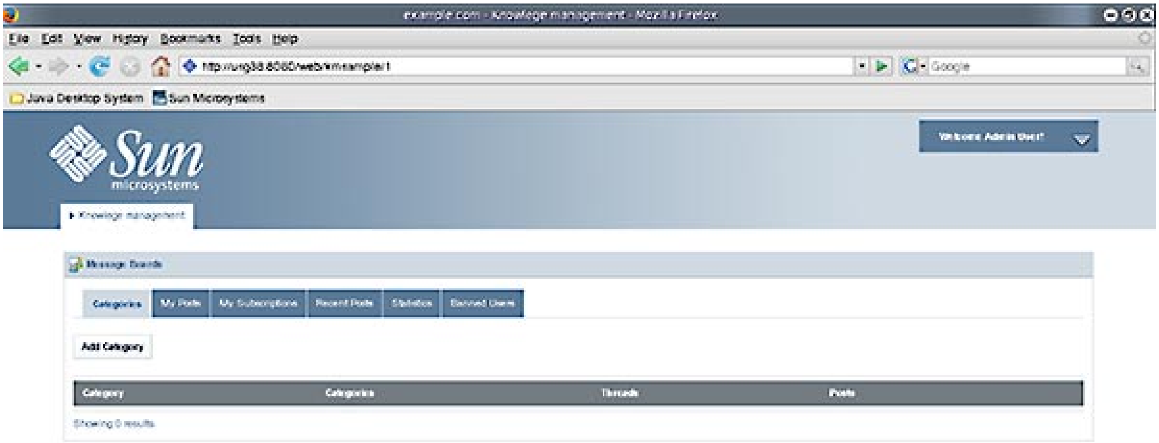


FIGURE 7–26 Staging Communities

6 **To view the staged page, choose Staging –> View Staged Page from the welcome menu.**

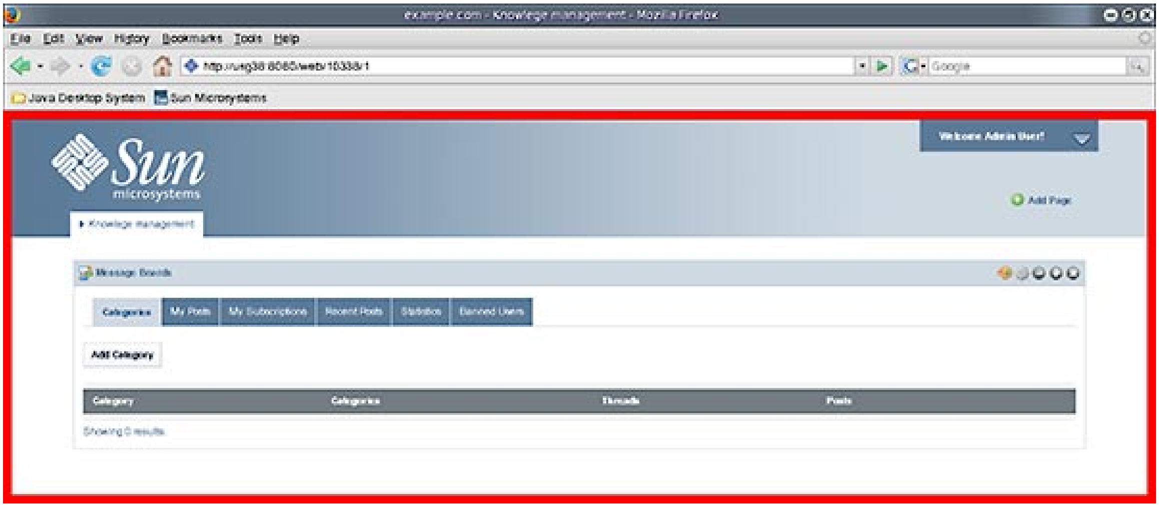
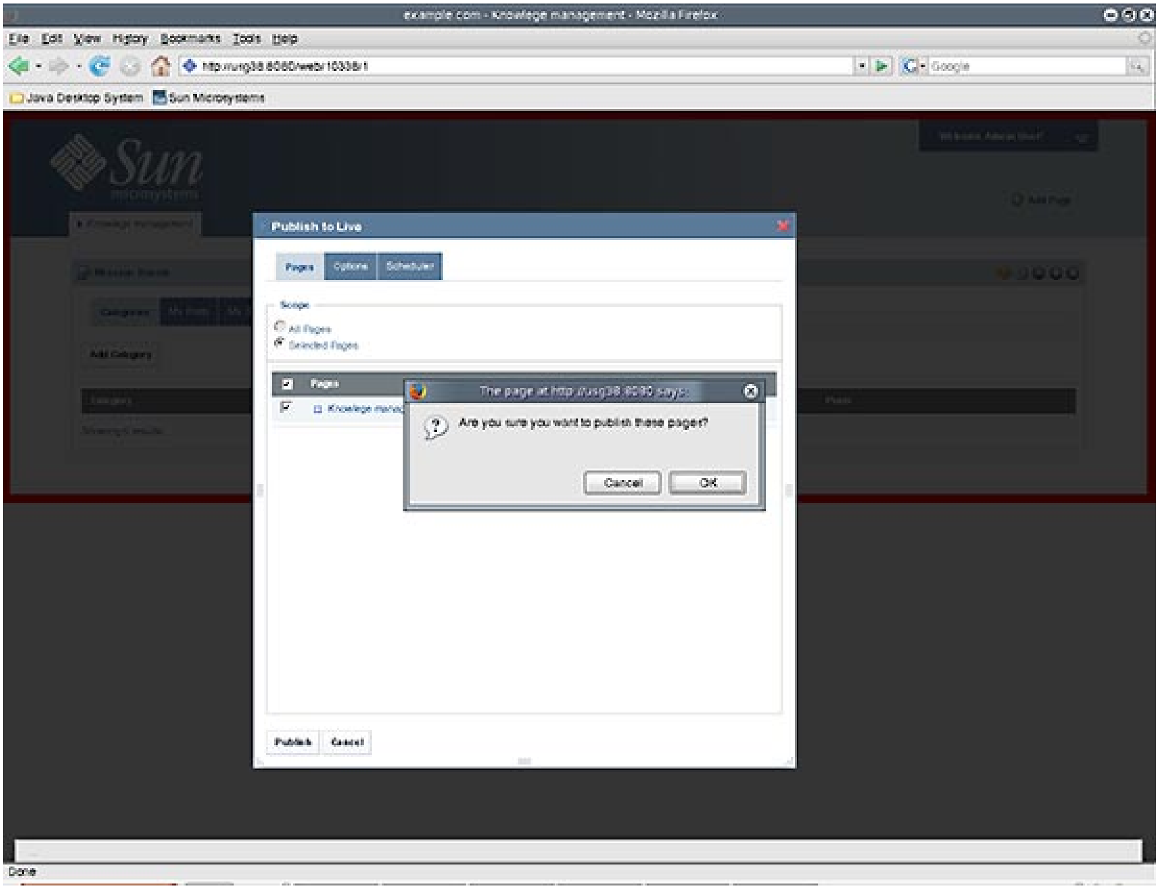


FIGURE 7–27 Staging Communities

7 **To publish the page to live, choose Staging —> Publish to live from the welcome menu and follow the UI instructions.**


example.com - Knowlege management - Mozilla Firefox
File Edit View History Bookmarks Tools Help
http://usg38:8080/web/10338/1
Google
Java Desktop System
Sun Microsystems
Publish to Live
Pages
Options
Scheduler
Publish
Cancel
Cancel
OK
Done

Scope

○ All Pages
● Selected Pages

☑	Pages
☑	Knowlege manag

The page at http://usg38:8080 says:

Are you sure you want to publish these pages?

FIGURE 7–28 Staging Communities

8 **To view the live page, choose Staging —> View Live Page from the welcome menu.**

Activating Workflow

In the above described procedure, when you select the *Activate Staging* option, the *Activate Workflow* option appears. If you select the Activate Workflow option, you are allowed to define a Workflow to the Community or the Organization.

Enabling Crawling of Page URLs by Search Engines

You can enable crawling of a page url by a search engine for the pages on Communities and Organizations.

The below procedure discusses enabling crawling of a page url on the Collaboration Sample Community. You can enable crawling for the pages on any other Community or Organization by following the similar procedure.

▼ To Enable Crawling of Page URLs by Search Engines

1 **Login to Project WebSynergy as Admin User.**

2 **Click on the Communities I Own tab in the My Communities portlet.**

3 **Click on the Actions button against the Collaboration Sample community and select Manage Pages from the menu.**

4 **Navigate to Settings —> Sitemap.**

You have *Google* and *Yahoo* links are available. By clicking on the Google link, you add your Sitemap to the Sitemaps to crawl on Google search engine. It works similarly for Yahoo. By clicking on the preview link, you can view the XML file which is sent to the webmaster on the selected serach engine for sitemap.

Virtual Hosting of Portlet Pages

WebSynergy supports virtual hosting of Communities and Organizations. Public and Private pages in a Community or an Organization have a unique common friendly URL. You have to provide separate host names for public and private pages for virtual hosting of public and private pages on separate machines. In the below example, we consider a simple use case of virtual hosting on the same machine by providing a convenient friendly URL.

▼ To Virtual Host a Page for a Community

1 **Login to Project WebSynergy as Admin User.**

2 **Click on the Organizations tab in the Enterprise Admin portlet.**

All existing Organizations are listed.

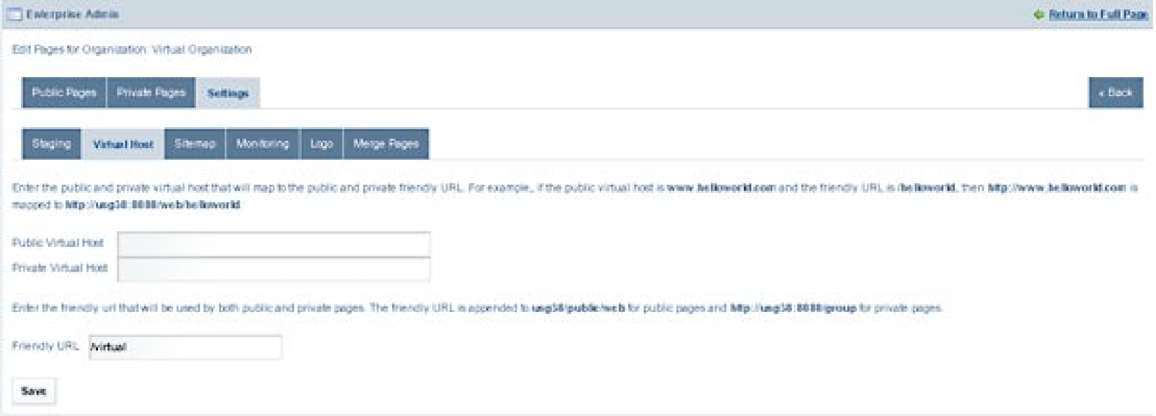
3 **Click on the Actions button against a selected Organization and choose Manage Pages from the menu.**

4 **Click on the settings tab.**

5 **Click on the Virtual Host tab.**

6. **Specify a Friendly URL.**

   In our example, 'virtual' is the friendly URL.



Enter the public and private virtual host that will map to the public and private friendly URL. For example, if the public virtual host is **www.helloworld.com** and the friendly URL is **/helloworld**, then **http://www.helloworld.com** is mapped to **http://usg38:8080/web/helloworld**

Public Virtual Host

Private Virtual Host

Enter the friendly url that will be used by both public and private pages. The friendly URL is appended to **usg38/public/web** for public pages and **http://usg38:8080/group** for private pages.

Friendly URL: /virtual

FIGURE 7–29 Virtual hosting

7. **click Save.**

8. **To view the public page, click on the Public Pages tab.**

9. **Click on View Pages button.**

   The public page for the Organization opens in a separate browser with its virtual hosting URL.

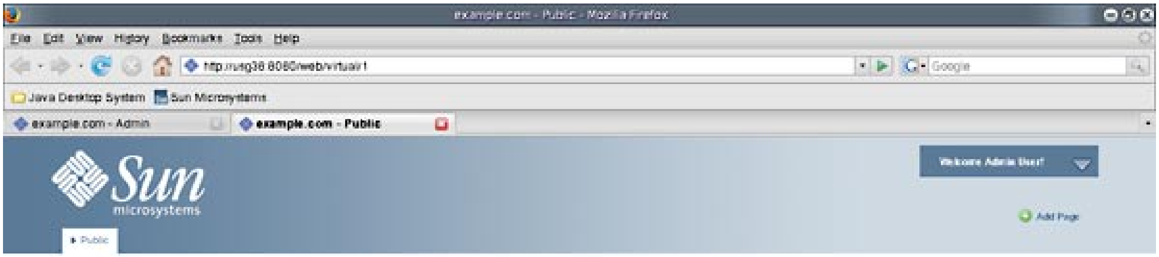


FIGURE 7–30 Virtual hosting

Using the Mail Portlet

To be done.

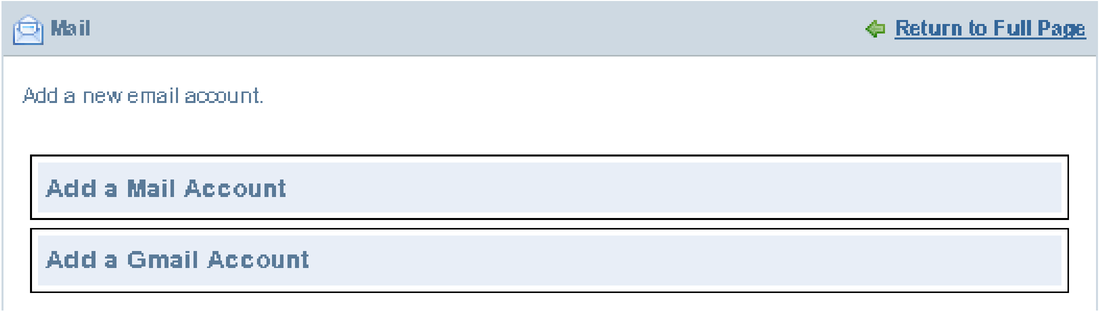
▼ To Configure Email Accounts

1 **Login to Project WebSynergy as Admin User.**

2 **Choose Add Applications from the welcome menu and add Mail portlet to your page.**

3 **Click on the Configure email accounts link.**

You are prompted to add mail accounts.



Add a new email account.

FIGURE 7–31 Adding Email Accounts

It is easy to add Gmail account as you can enable IMAP in a few simple steps. Follow the below procedure to enable IMAP for your Gmail account:

a. **Sign in to Gmail.**

b. **Click Settings at the top of any Gmail page.**

c. **Click Forwarding and POP/IMAP.**

d. **Select Enable IMAP.**

e. **Click Save Changes.**

4 **Click on Add a Gmail Account.**

5 **Enter your user name and password and click Save.**

Your email account is added.

6 **Click on the Check your email link.**

You can read the mails and folders in your Gmail account and can compose new mails.

Exporting and Importing of Portlet Data

You can take a backup for the data that you have on a portlet by exporting it to the disk. When you upgrade the WebSynergy installation on your machine, or when you reinstall WebSynergy due to some other reason, you can import the data back to your portlet. Consider the example of exporting the date available on the Calendar portlet.

The following is the scenario involved:

1. Add the Calender portlet to your page. The portlet is available by default on some sample user pages.
2. Added an event to the Calendar.
3. Export the date to your disk. The data is exported to the disk in `.ics` format. You can open the file in any text editor and can view the information related with the event. The ICS file can be used to import the event information when needed.

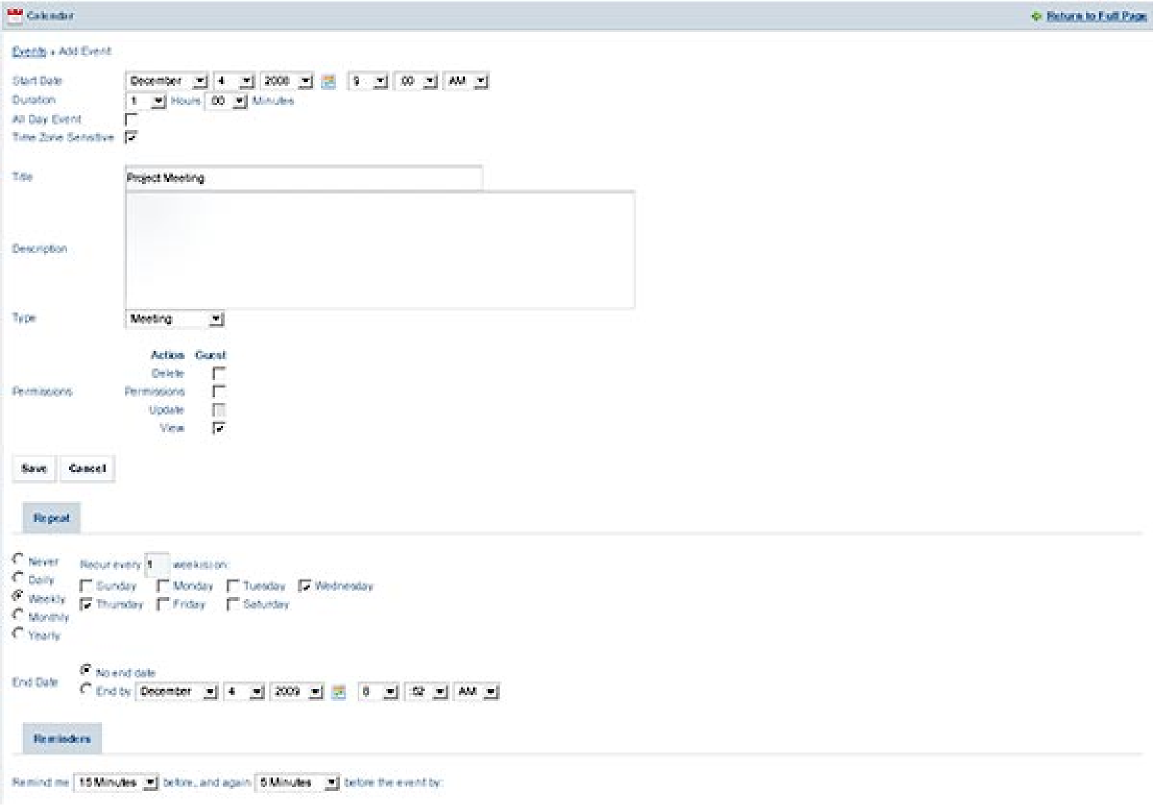
Note – ICS is a data format store calendar information in a text file on the Internet. ICS stands for iCalendar Calendar Data. ICS files have `.ics` extension.

▼ To Demonstrate Exporting the Data on Calendar Portlet

1 **Login to Project WebSynergy as Paul Tester.**

2 **Click on the Add Event button.**

In this example, we are adding a weekly event called project meeting. You can modify permissions for the guest user by clicking on the *Permissions* link.



Calendar — Return to Full Page

Events » Add Event

Start Date: December 4 2008 9 : 00 AM
Duration: 1 Hours 00 Minutes
All Day Event: □
Time Zone Sensitive: ☑
Title: Project Meeting
Description:
Type: Meeting

Permissions:

Action	Guest
Delete	□
Permissions	□
Update	□
View	☑

Save Cancel

Repeat

○ Never
○ Daily
● Weekly
○ Monthly
○ Yearly

Recur every 1 week(s) on:
□ Sunday □ Monday □ Tuesday ☑ Wednesday
☑ Thursday □ Friday □ Saturday

End Date:
● No end date
○ End by December 4 2009 8 : 52 AM

Reminders

Remind me 15 Minutes before, and again 5 Minutes before the event by:

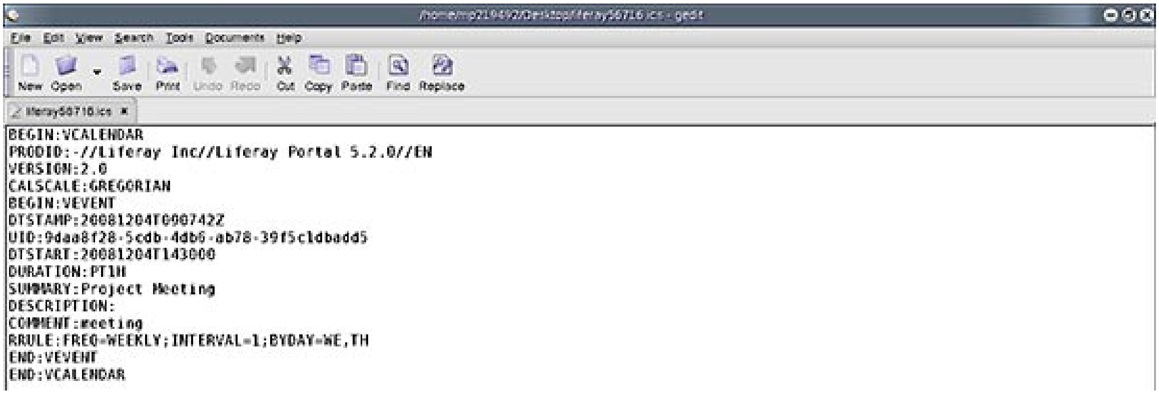
FIGURE 7–32 Adding an event

3. **Click Save.**

4. **Click on the Export/Import tab.**

5. **Click Export and save the file to the disk.**

   You can open the file by using any text editor. The file includes the event description. You can import the event information by importing the file to the Calendar portlet.


/home/mp219492/Desktop/liferay56716.ics - gedit
File Edit View Search Tools Documents Help
New Open Save Print Undo Redo Cut Copy Paste Find Replace
liferay56716.ics

```
BEGIN:VCALENDAR
PRODID:-//Liferay Inc//Liferay Portal 5.2.0//EN
VERSION:2.0
CALSCALE:GREGORIAN
BEGIN:VEVENT
DTSTAMP:20081204T090742Z
UID:9daa8f28-5cdb-4db6-ab78-39f5c1dbadd5
DTSTART:20081204T143000
DURATION:PT1H
SUMMARY:Project Meeting
DESCRIPTION:
COMMENT:meeting
RRULE:FREQ=WEEKLY;INTERVAL=1;BYDAY=WE,TH
END:VEVENT
END:VCALENDAR
```

FIGURE 7–33 The event data

Enabling Sharing of a Portlet

You can enable sharing of a portlet on any website, on Facebook, and by Friends.

- “Sharing a Portlet With Other Websites” on page 258
- “Sharing a Portlet With Facebook” on page 261
- “Sharing a Portlet With Friends” on page 262

Sharing a Portlet With Other Websites

Consider the example of sharing the Calendar portlet on a website created by you. The following is the scenario involved:

1. Click on the *Configuration* button among the edit controls for a portlet.
2. Click on the *Sharing* tab.

   Any Website, *Facebook* and *Friends* are the three tabs within the Sharing tab.
3. Copy the code in the Any Website tab.
4. Select the Allow users to add <portlet> to any website option and click Save.
5. Paste the copied code into the html source of any website and save the page. The portlet is added to the website.

▼ To Share a Portlet With Other Websites

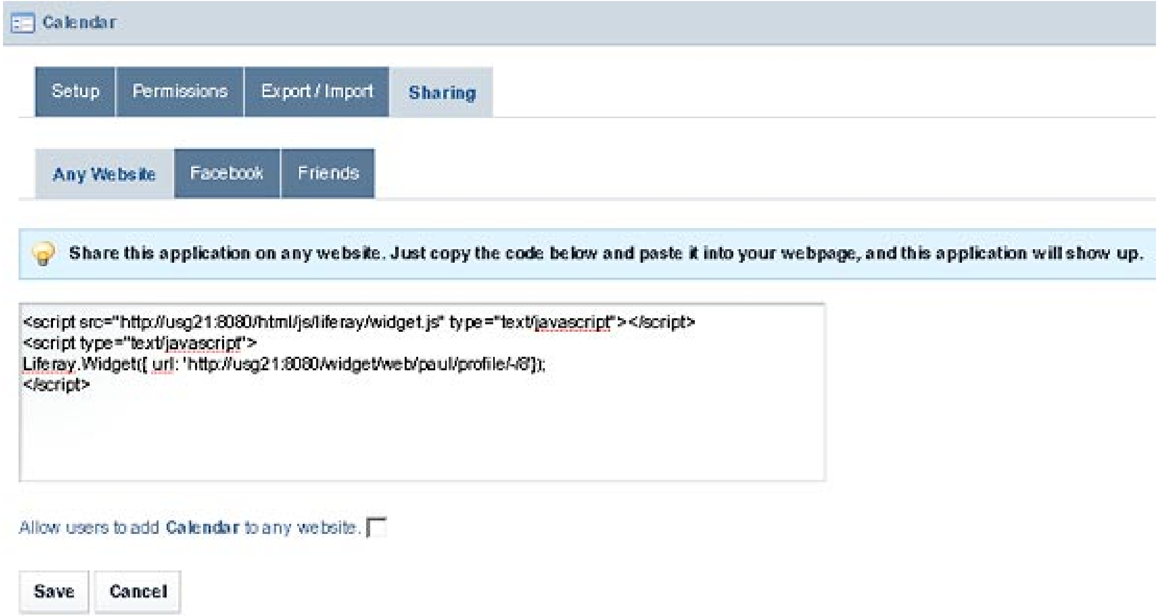
1 **Login to Project WebSynergy as any user of your choice.**

In this example, login as Paul Tester.

2. **Click on the Configuration button among the edit controls for the portlet.**

   In this example, click on the Configuration button among the edit controls for the Calendar portlet.

3. **Click on the Sharing tab.**



Share this application on any website. Just copy the code below and paste it into your webpage, and this application will show up.

```
<script src="http://usg21:8080/html/js/liferay/widget.js" type="text/javascript"></script>
<script type="text/javascript">
Liferay.Widget({ url: 'http://usg21:8080/widget/web/paul/profile/-/8'});
</script>
```

Allow users to add Calendar to any website. □

FIGURE 7–34 The code for sharing the portlet on any website

4. **Copy the code for sharing the portlet on any website.**

5. **Select the Allow users to add Calendar to any website option.**

6. **Click Save.**

7. **Open the source code of any website and copy the sharing code into it, and save changes to add the portlet to the website.**

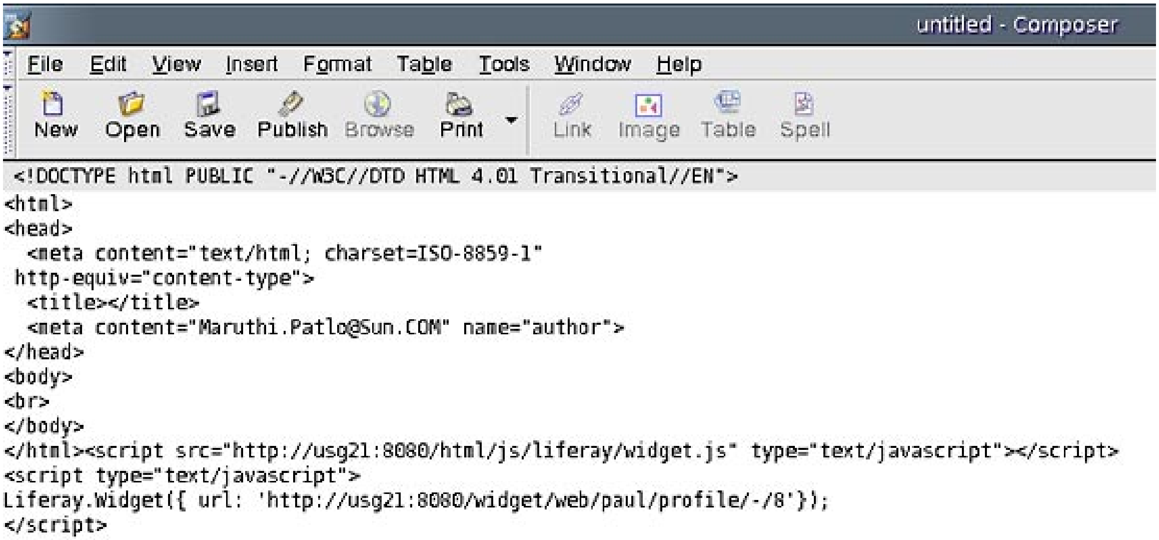
   As a simple example for this, do the following:

   a. **Open the Mozilla Classic browser**

   b. **Click Edit Page from the file menu.**

c. **Click on HTML Source.**

d. **Paste the copied sharing code into the page.**



```
 <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
  <meta content="text/html; charset=ISO-8859-1"
 http-equiv="content-type">
  <title></title>
  <meta content="Maruthi.Patlo@Sun.COM" name="author">
</head>
<body>
<br>
</body>
</html><script src="http://usg21:8080/html/js/liferay/widget.js" type="text/javascript"></script>
<script type="text/javascript">
Liferay.Widget({ url: 'http://usg21:8080/widget/web/paul/profile/-/8'});
</script>
```

FIGURE 7–35 Adding the sharing code to the source code of a website

e. **Click Save.**

f. **A dialog box requesting for a page title appears. Type a title and click OK, and save the page your desktop.**

In this example, we are naming the page as *Calendar Sharing*.

g. **Open the page in a web browser. You can notice that the Calendar portlet is added to the page.**

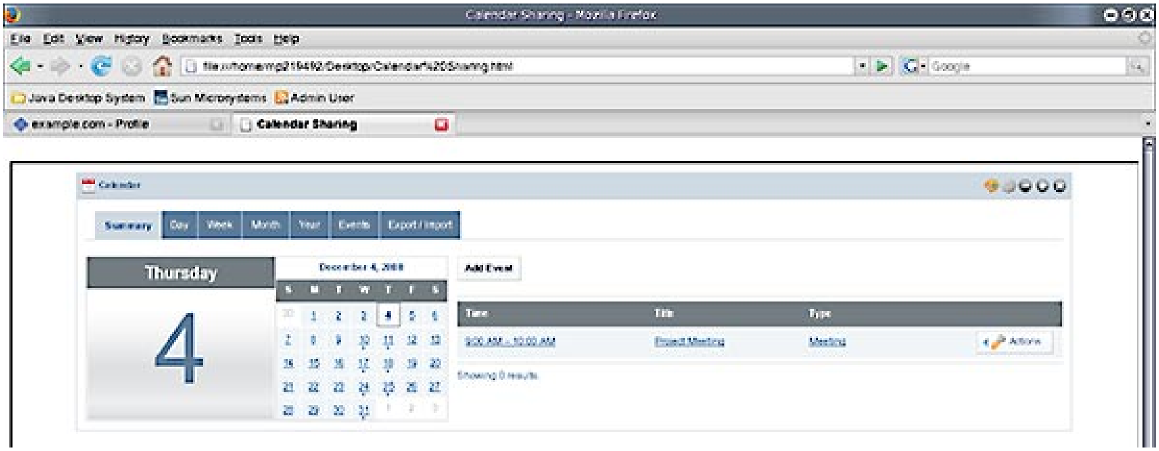


FIGURE 7–36 The portlet added to the website

Sharing a Portlet With Facebook

The following is the scenario involved in sharing a portlet with Facebook:

1. Click on the Configuration button among the edit controls for the portlet.
2. Click on the *Sharing* tab.
3. Click on *Facebook* under Sharing.
4. Click the *Get the API key and canvas page URL from Facebook*, and follow the instructions to get the API key and a canvas page URL for the portlet.
5. Specify the API key and the canvas page URL and click Save.
6. You can access the portlet on Facebook by using the canvas page URL.

▼ To Share a Portlet With Facebook

1 **Login to Project WebSynergy as any user of your choice.**

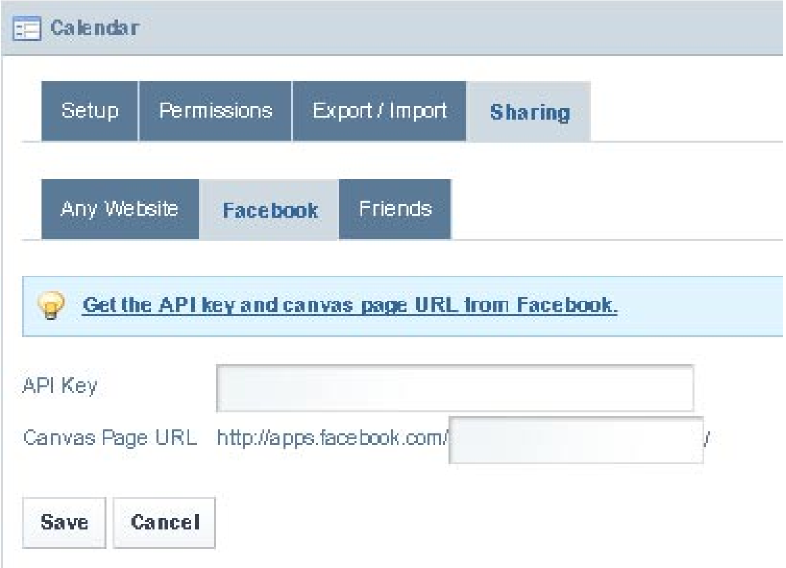
In this example, login as Paul Tester.

2 **Click on the Configuration button among the edit controls for the portlet.**

In this example, click on the Configuration button among the edit controls for the Calendar portlet.

3 **Click on the Sharing tab.**

4 **Click on Facebook under Sharing.**



Calendar

Get the API key and canvas page URL from Facebook.

API Key

Canvas Page URL http://apps.facebook.com/ /

FIGURE 7–37 Sharing on Facebook

5 **Click on 'Get the API key and canvas page URL from Facebook' link and follow the instructions to get the API key and canvas page URL from Facebook.**

6 **specify the API key and the canvas page URL and click Save.**

7 **Access the portlet on Facebook by using the canvas page URL.**

Sharing a Portlet With Friends

The Friends portlet lists the WebSynergy users who are registered as your friends. You can authorize your friends to access a portlet which is on your page.

You need to perform the following steps to enable friends to access a portlet on your page:

1. Click on the Configuration button among the edit controls for the portlet.
2. Click on the *Sharing* tab.
3. Click on *Friends* under Sharing.
4. Select the *Allow users to share <portlet> with friends* option and click Save.

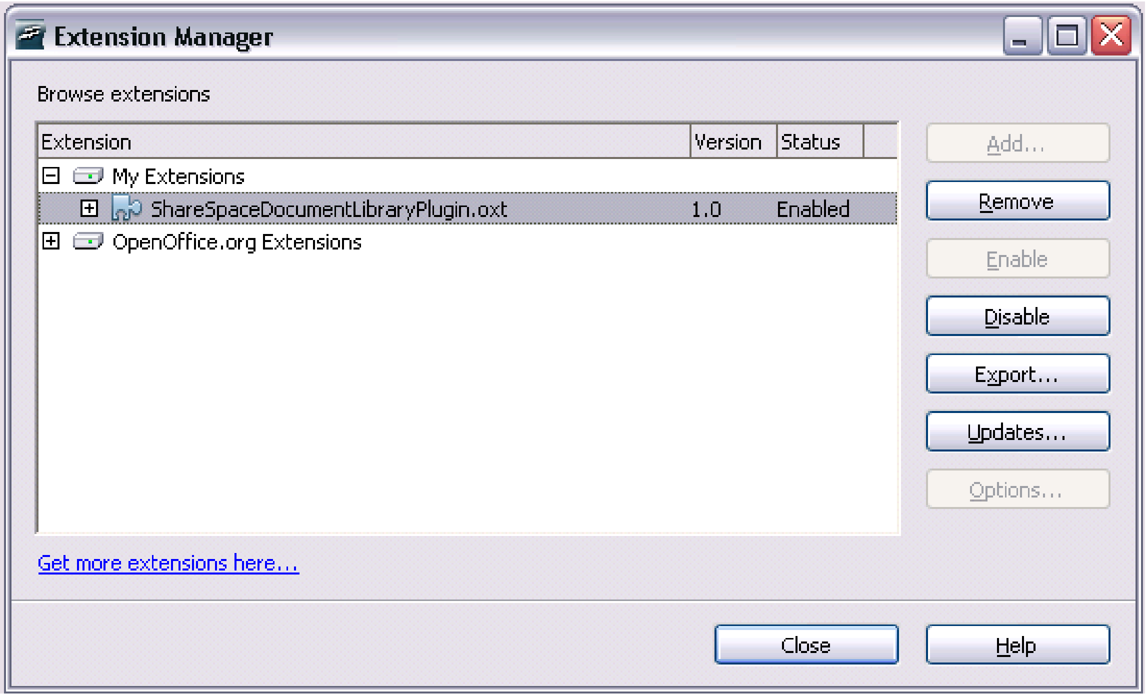
Using Document Library Plugins on OpenOffice

Document Library Plugins are helpful to establish an interface with WebSynergy from OpenOffice. Document Library Plugins are shipped along with the standard product. When you install the pulgins on OpenOffice, you will be able to interface will the CMS feature of WebSynergy. This enables you to use OpenOffice to edit the documents you have in the Document Library. Also, you can add documents to the Document Library.

Note – OpenOffice bundle can be downloaded freely form `http://www.openoffice.org/`.

▼ To Store a Document to CMS

1. **Open OpenOffice – Writer.**
2. **Choose Tools —> Extension Manager.**
3. **Click on the Add button and browse 'ShareSpaceDocumentLibraryPlugin.oxt'.**



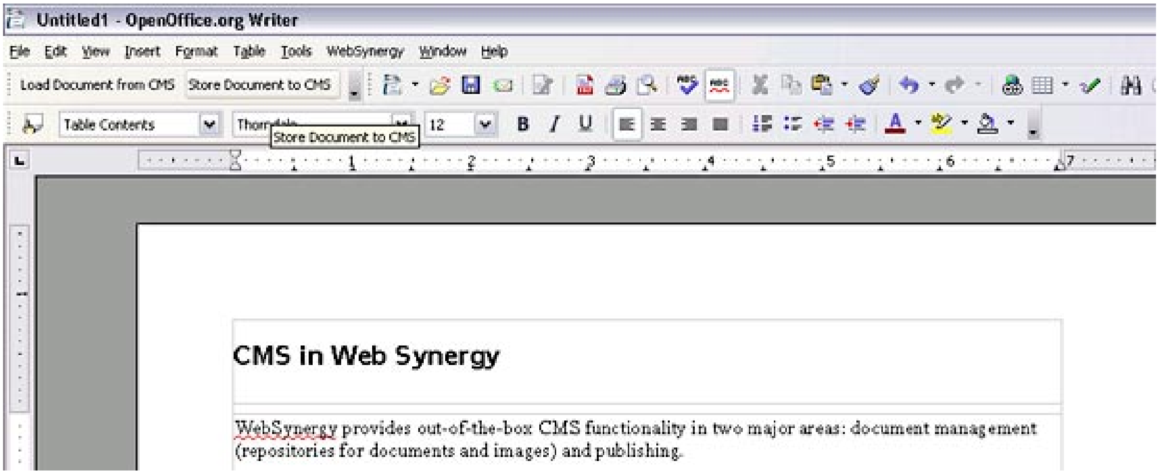
Extension	Version	Status
⊟ My Extensions		
⊞ ShareSpaceDocumentLibraryPlugin.oxt	1.0	Enabled
⊞ OpenOffice.org Extensions		

FIGURE 7–38 Installing the Document Library plugins

The Document Library plugins are added to your page. You can notice the *Load Document from CMS* and *Store Document to CMS* buttons on the toolbar.

4 **Write some content that you want to store on the CMS.**

5 **Click on the Store Document to CMS button to upload the document to CMS.**


Untitled1 - OpenOffice.org Writer
File Edit View Insert Format Table Tools WebSynergy Window Help
Load Document from CMS
Store Document to CMS
Table Contents
Thorndale
12
Store Document to CMS

CMS in Web Synergy

WebSynergy provides out-of-the-box CMS functionality in two major areas: document management (repositories for documents and images) and publishing.

FIGURE 7–39 Storing a document to Document Library

Using the Widget Consumer Portlet

You can embed a web widget by copying the embed-able javascript or flash code snippet and pasting it in the text box after clicking the Configuration button on the Widget Consumer portlet. On saving the changes to the configuration, you will be able to view the web widget from the portlet.

Widget Consumer portlet is an instanceable portlet, and it allows you to add any number of instances of the portlet to your page with each portlet instance embedding a separate widget. For example, in any YouTube link, you can find a code snippet named 'Embed' at the right top corner where you see the Subscribe button. You can copy paste the code to the Widget Consumer portlet to view the YouTube link.

The below procedure, describes an example of copying the snippet code for a web widget from the website `cricinfo.com` and using it on the Widget Consumer portlet.

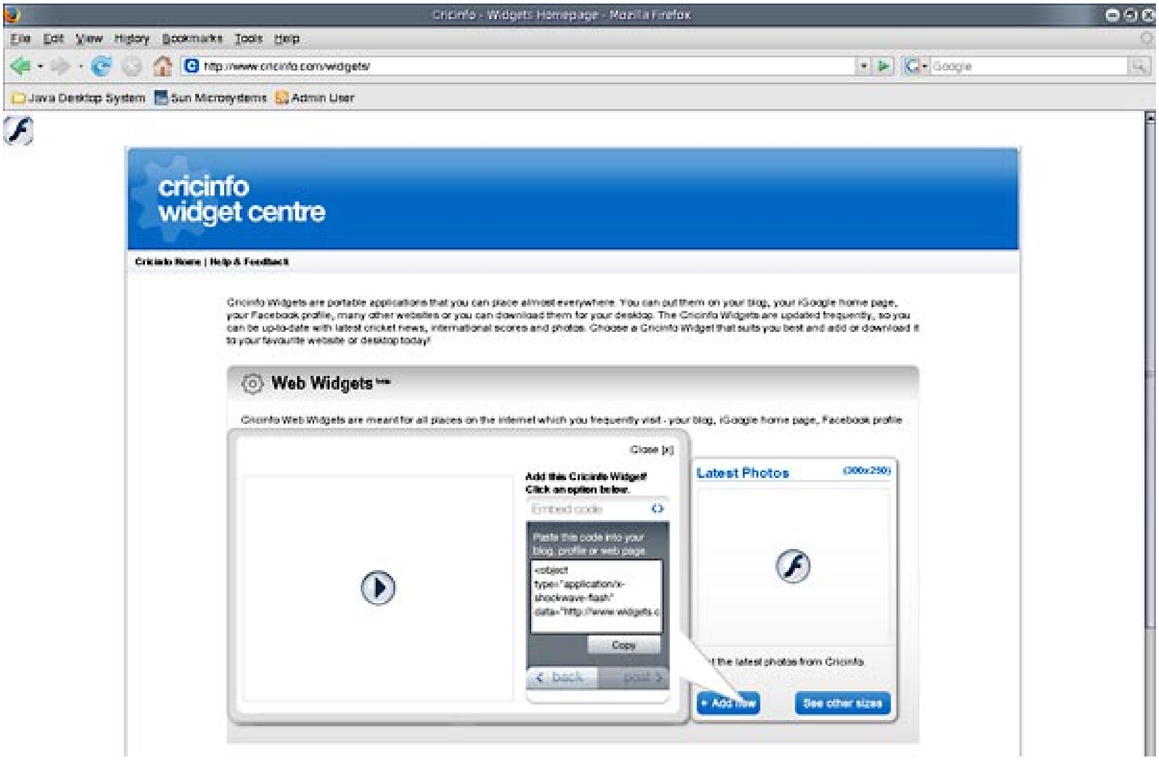
▼ To Configure the Widget Consumer Portlet

1. **Login to Project WebSynergy as any user of your choice.**

2. **Choose Add Application from the welcome menu and add Widget Consumer portlet to your page.**

3. **Copy the code snippet to embed a web widget.**

   In this example, copy the code snippet to embed a web widget from the site `cricinfo.com`.



Cricinfo Widgets are portable applications that you can place almost everywhere. You can put them on your blog, your iGoogle home page, your Facebook profile, many other websites or you can download them for your desktop. The Cricinfo Widgets are updated frequently, so you can be up-to-date with latest cricket news, international scores and photos. Choose a Cricinfo Widget that suits you best and add or download it to your favourite website or desktop today!

Cricinfo Web Widgets are meant for all places on the internet which you frequently visit - your blog, iGoogle home page, Facebook profile

Paste this code into your blog, profile or web page.

```
<object
type="application/x-
shockwave-flash"
data="http://www.widgets.c
```

...t the latest photos from Cricinfo.

FIGURE 7–40 Copying the code snippet to embed a widget

4. **Click on the Configuration button on the portlet.**

5. **Paste the widget code into the text box.**

6. **Click Save.**

   You can view the widget from the Widget Consumer portlet.

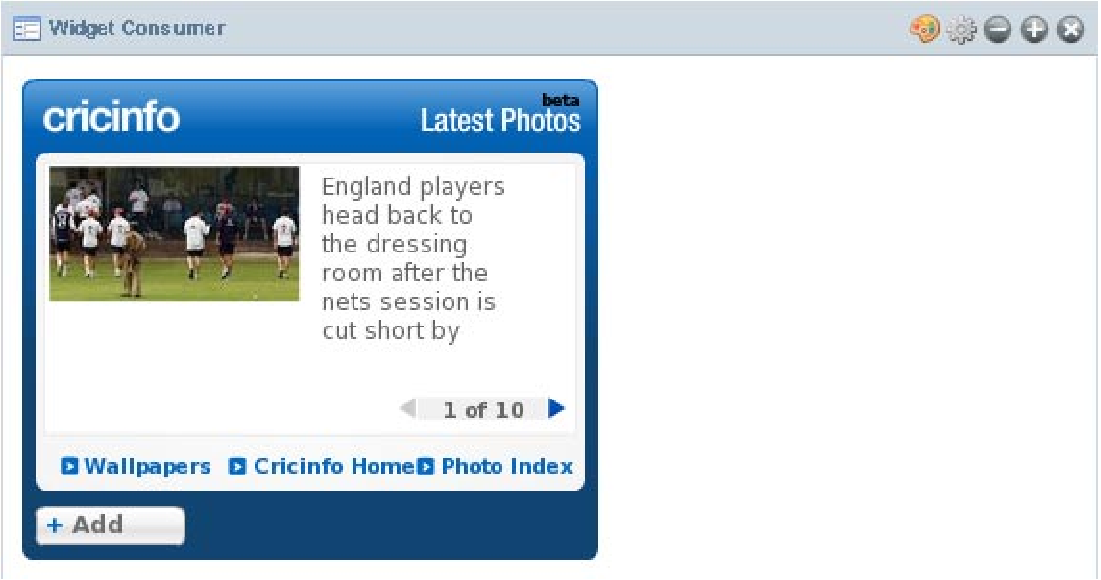


FIGURE 7–41 The widget added to Widget Consumer

Tagging in WebSynergy

The tagging system in WebSynergy helps to group blogs, wikis, and articles. Admin user can create tag sets and tags, and can define permissions to use tags by using the *Tags and Categories* portlet. Tags follows a kind of taxonomy and classification, as the admin user can define tag sets and tags within them. Users will have the freedom to create their own tags or to use the existing tags. The basic process of using the tagging system is:

- The admin user creates tags and tag sets, and defines permissions for usage.
- The user applies tags for his blogs, wikis, or articles. He can select the tags that he want to apply, or can define his own tags.
- Any content that is tagged is grouped or aggregated.

CHAPTER 8

Web Services for Remote Portlets

WSRP 2.0 is an OASIS standard that simplifies integration of remote applications and content into portals. The WSRP standard defines presentation-oriented, interactive web services with a common, well-defined interface and protocol for processing user interactions and for providing presentation fragments suited for mediation and aggregation by portals as well as conventions for publishing, finding and binding such services.

Because the WSRP interfaces are common and well-defined, all web services that implement the WSRP standard plug into all WSRP compliant portals - a single, service-independent adapter on the portal side is sufficient to integrate any WSRP service. As a result, WSRP is the means for content and application providers to provide their services to organizations running portals with no programming effort required.

See the WSRP 2.0 standard for more information:

`http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=wsrp`

The implementation of the WSRP 2.0 standard inProject WebSynergy includes both the WSRP consumer and the WSRP producer. The WSRP producer implementation supports publishing JSR286 portlets for use by a remote WSRP consumer. The JSR286 portlets are deployed locally on a portal server. These portlets can be published by an instance of the WSRP producer.

WSRP for Project WebSynergy

- "Adding WSRP Portlets to the Page" on page 269
- "WSRP Administration" on page 271

Adding WSRP Portlets to the Page

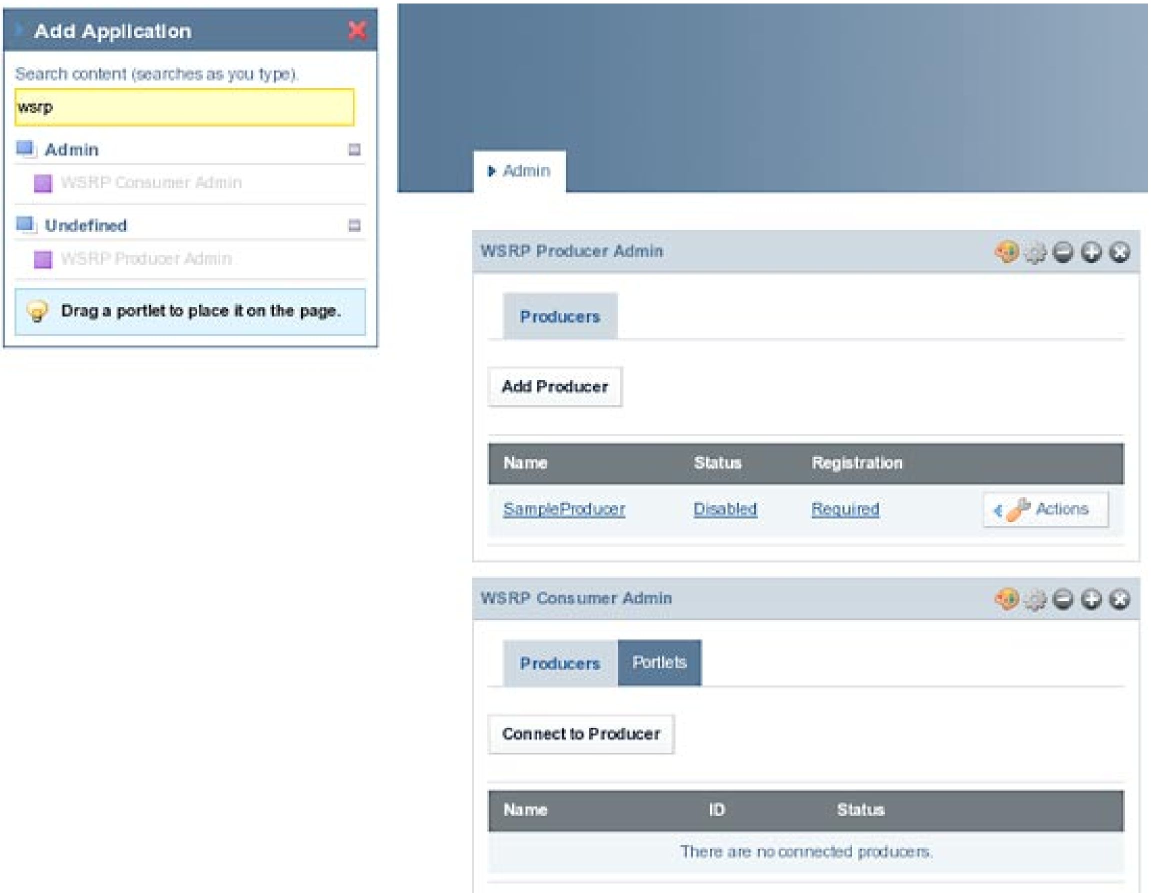
You need to add the *WSRP Producer Admin* and *WSRP Consumer Admin* portlets to your page. These two portlets hold the WSRP functionality.

Generally, in any real life scenario, WSRP Producer and WSRP Consumer are on different machines, with WSRP Consumer running on a remote machine. For the sake of simplicity, we are using WSRP Producer Admin and WSRP Consumer Admin portlets from a single WebSynergy installation.

▼ To add WSRP Portlets to the Page

1 **Login to WebSynergy as Admin user.**

2 **Click on Add Applications from the welcome menu, and type 'wsrp' in the search field.**



Search content (searches as you type): wsrp

Name	Status	Registration	
SampleProducer	Disabled	Required	Actions

Name	ID	Status
There are no connected producers.		

FIGURE 8–1 To add WSRP portlets to your page

The *WSRP Consumer Admin* and *WSRP Producer Admin* are listed.

3 Add the portlets to your page, by clicking on the Add button.

WSRP Administration

The WSRP Producer Admin and WSRP Consumer Admin portlets provide WSRP Administration functionality.

- "Creating a WSRP Producer" on page 271
- "Creating a WSRP Consumer" on page 273
- "Creating WSRP Producers and Consumers Without Registration" on page 275
- "Installing Remote/WSRP Portlets" on page 276

Creating a WSRP Producer

The WSRP Producer Admin portlet could be used for creating and managing WSRP Producers. There exists the default producer called *SampleProducer* which is disabled by default.

What is a WSRP Producer?

A WSRP Producer offers locally deployed portlets for remote deployment. A remotely deployed WSRP Consumer Admin portlet can connect to the Producer, and can use the portlets offered by the Producer.

A portal can host multiple producers. The consumer can import remote portlets offered by a producer. Based on the portlets that you want to provide to WSRP consumers, you may create one or more producers. A producer can support registration or it does not require registration. If a producer supports registration, then consumers must register to work with the producer.

Inband and Outband Registration

Registration is used to build a technical or business relationship between the consumer and the producer. While creating a producer, you can define any one of the following registration mechanisms: Inband registration or Outband registration.

If the producer requires registration and enabled for Inband registration, the consumer can provide the details through WSRP interface and register with the producer. Consumer is also provided an option to register through Outband communication. That is, consumer can provide the registration handle obtained through Outband communication.

If the producer requires registration and enabled for Outband registration, the consumer should obtain the registration handle through Outband communication and provide the registration handle during registration. Outband registration happens with manual intervention such as phone calls, email, and so on. For a producer that supports Outband registration, the producer gets the details about the consumer through Outband communication, and it creates a registration handle for the consumer. The registration handle is communicated to the consumer through Outband communication.

Producers Without Registration

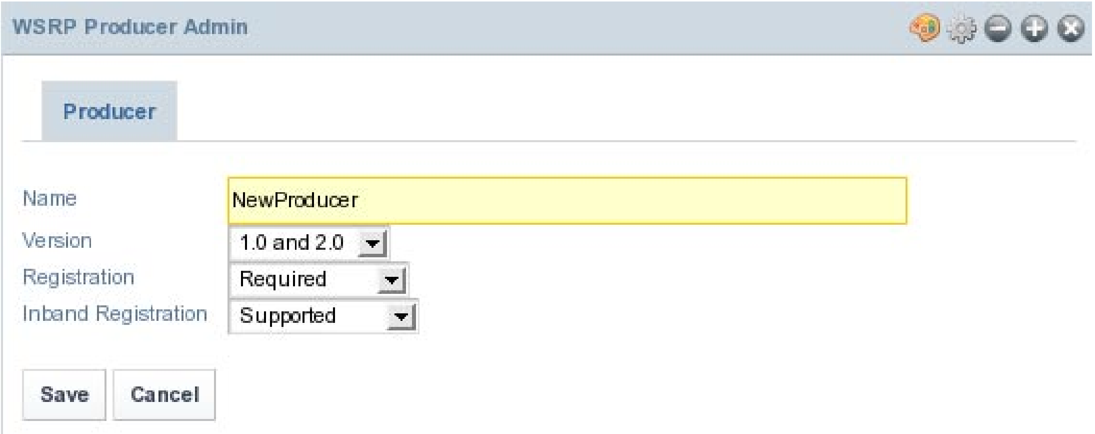
For a producer that does not require registration, consumer is not required to enter any information or get any information through Outband communication. In this case, the consumer can not customize (or edit) the portlets offered by the producer. The producer that does not support registration provides Read-Only portals to the consumers.

▼ To Create and Enable a WSRP Producer

1. **Click the Add Producer button in the WSRP Producer Admin portlet.**

   Specify a name for the Producer. In our example, we are creating a Producer called *NewProducer*. This screen also provides option to export a version of wsrp service (choose both 1.0 and 2.0) for exporting both version 1.0 and version 2.0 service.

   Note – The name of the Producer or a Consumer must not contain empty spaces or any special characters.



Name: NewProducer
Version: 1.0 and 2.0
Registration: Required
Inband Registration: Supported

FIGURE 8–2 Creating a new WSRP Producer

2. **Click Save.**

   The Producer is created, but the new Producer will be disabled by default.

3. **To enable the WSRP Producer:**

   a. **Click the producer link in theWSRP Producer Admin portlet.**

   In this example, click *NewProducer*.
   The producer opens.

b. **Move some portlets from Unpublished to Published.**

c. **Click Save.**

The Producer is saved.

d. **Click the Disabled link.**

This opens the Producer again. Select Enabled from the Status list and click Save. The producer is enabled.

Note – By default the above created WSRP producer exports both WSRP version 1.0 and WSRP version 2.0 services in the exported WSDL. On checking the WSDL URL, both version 1.0 and version 2.0 services could be observed.

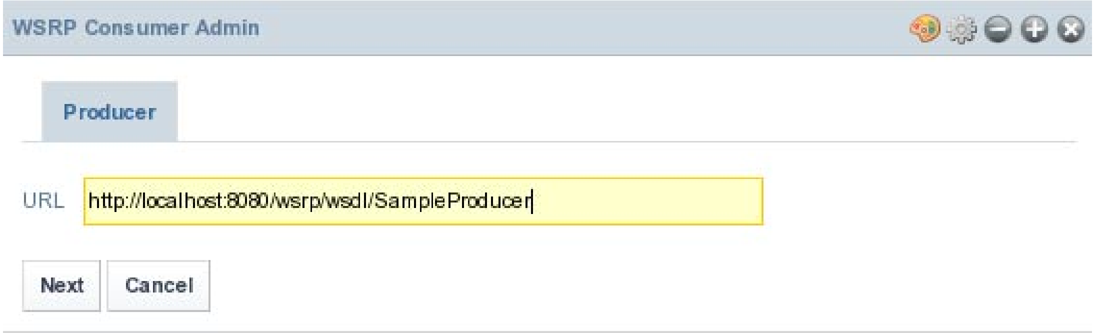
Creating a WSRP Consumer

To communicate with the portlets offered by the producer, a consumer needs to add a configured producer. The WSRP Consumer Admin portlet could be used for creating and managing WSRP Consumers. To create a consumer a producer WSDL URL is required. You can choose the WSDL URL from the "SampleProducer" created in the above step.

▼ To create a WSRP Consumer

1 **Click on the 'Connect to Producer' button from the 'Producers' tab in the 'WSRP Consumer Admin' portlet.**

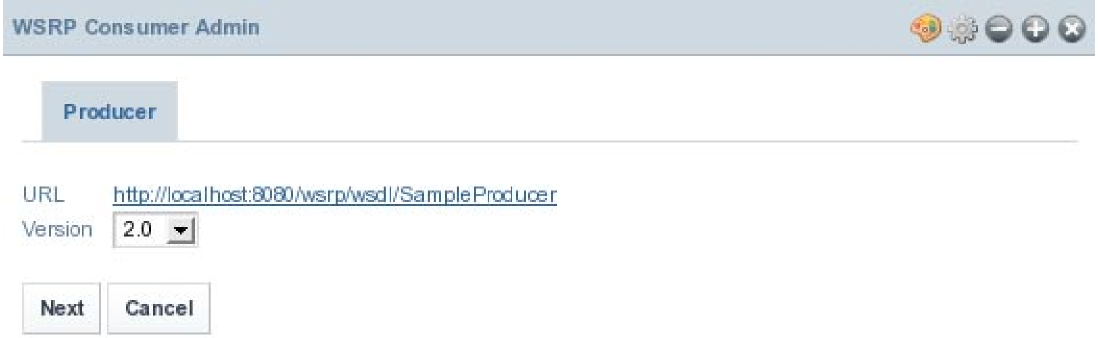
2 **Provide the WSRP Producer WSDL URL and click Next.**



URL http://localhost:8080/wsrp/wsdl/SampleProducer

FIGURE 8–3 Creating a WSRP Consumer

The next screen prompts to choose the WSRP version. Choose Version *2.0* and click *Next*.

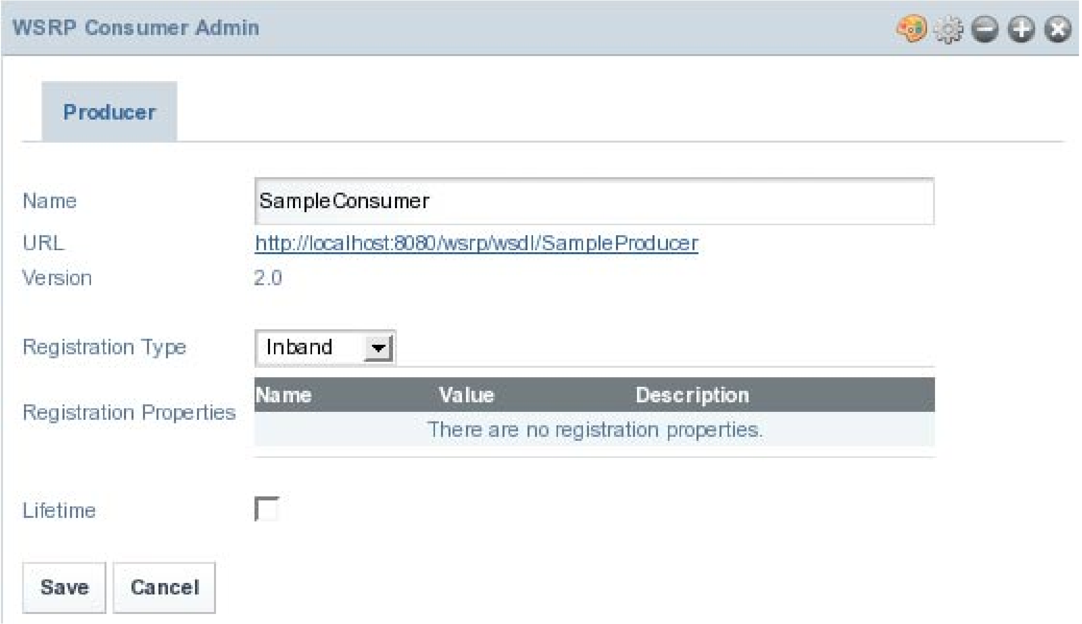


URL http://localhost:8080/wsrp/wsdl/SampleProducer

Version 2.0

FIGURE 8–4 Creating a WSRP Consumer

The next screen prompts you to select a registration type. *Inband* and *Outband* are the two registration types. It is optional to specify *Name* and *Lifetime* for the consumer.



Name SampleConsumer

URL http://localhost:8080/wsrp/wsdl/SampleProducer

Version 2.0

Registration Type Inband

Registration Properties

Name	Value	Description
There are no registration properties.		

Lifetime □

FIGURE 8–5 Creating a WSRP Consumer

In case of Outband registration, you have to specify a registration handle. Select Inband registration and click *Save*. The consumer is created.

Creating WSRP Producers and Consumers Without Registration

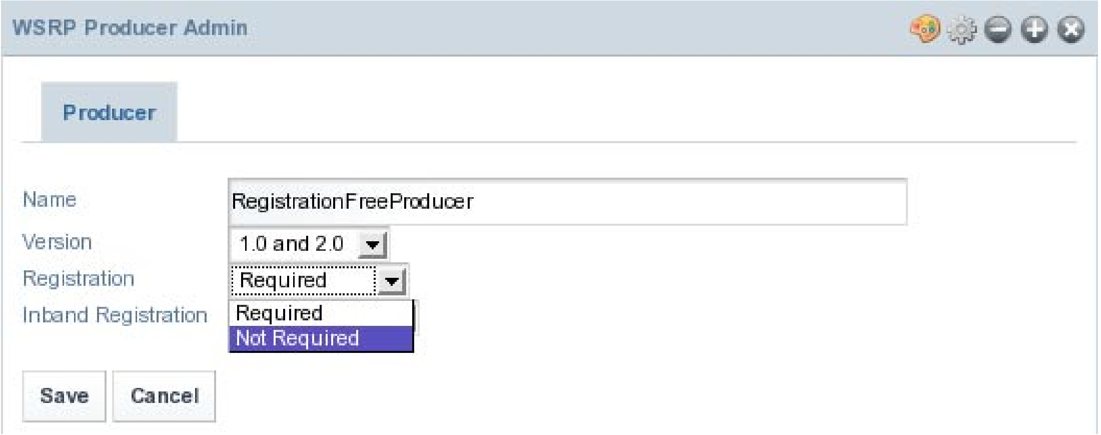
If the producer does not require registration, the consumer is not required to enter any details while adding a configured producer.

- "To Create a Producer without Registration" on page 275
- "To Create a Consumer without Registration" on page 275

▼ To Create a Producer without Registration

1 **Click on the Add Producer button in the WSRP Producer Admin portlet.**

Specify a name for the producer. Select *Not Required* from the list for *Registration*.



WSRP Producer Admin

Producer

Name: RegistrationFreeProducer

Version: 1.0 and 2.0

Registration: Required

Inband Registration

FIGURE 8–6 Creating a Producer without Registration

2 **Click Save.**

The Producer is created, but the new Producer will be disabled by default. Enable the Producer. See, "To Create and Enable a WSRP Producer" on page 272.

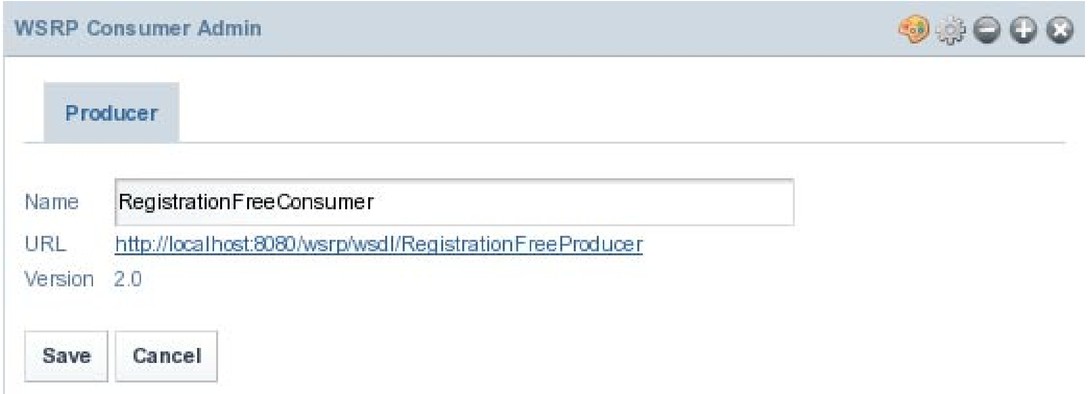
▼ To Create a Consumer without Registration

1 **Click on the 'Connect to Producer' button from the 'Producers' tab in the 'WSRP Consumer Admin' portlet.**

2 **Provide the WSDL URL of a Producer without registration click Next.**

3 **The next screen prompts to choose the WSRP version. Choose Version *2.0* and click *Next*.**

4 **Specify a name for the Consumer and Click Save.**



WSRP Consumer Admin

Producer

Name RegistrationFreeConsumer

URL http://localhost:8080/wsrp/wsdl/RegistrationFreeProducer

Version 2.0

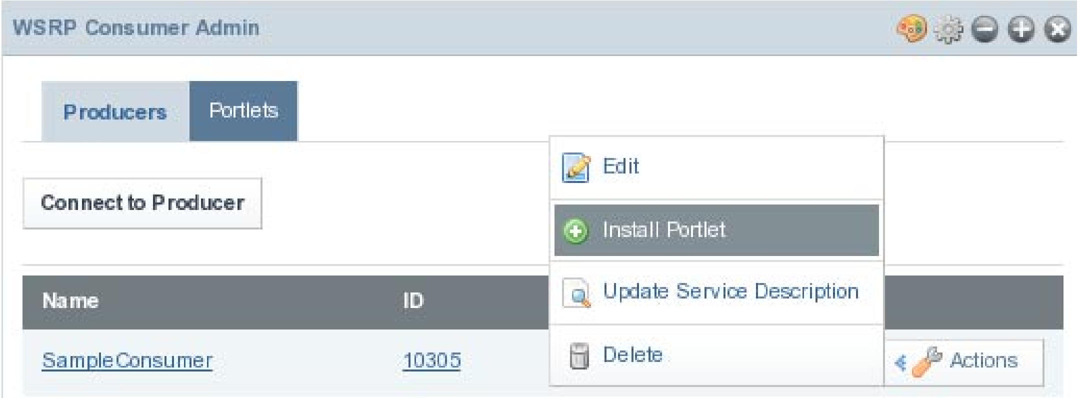
FIGURE 8–7 Creating a Consumer without Registration

Installing Remote/WSRP Portlets

To be done.

▼ To install a Remote/WSRP Portlet

1 **Click on the 'Actions' button against a Consumer from the 'WSRP Consumer Admin' portlet, and select 'Install Portlets' from the menu.**

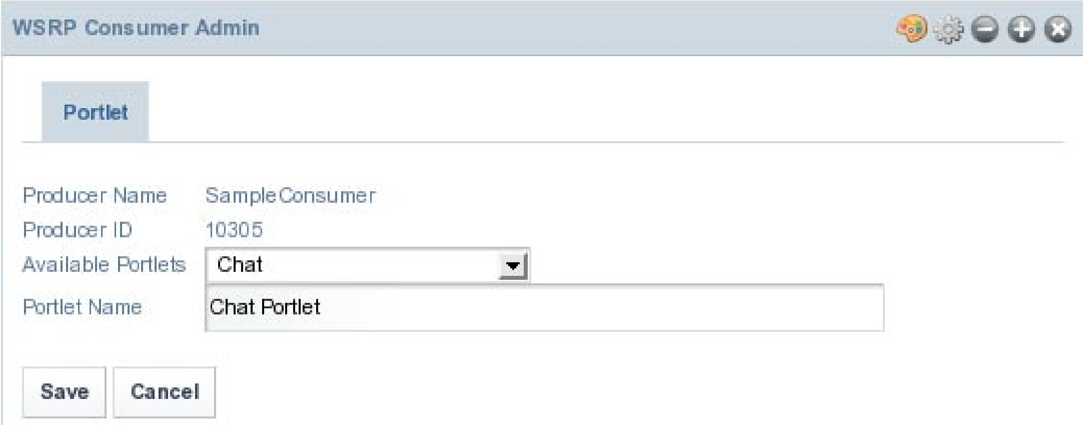


Name	ID
SampleConsumer	10305

FIGURE 8–8 Installing Remote/WSRP Portlets

The *Available Portlets* list displays all the portlets available with the Producer. Select a portlet from the Available Portlets list, and Specify a name for the portlet.

Note – Make sure to specify a name for the portlet. If you are not specifying a name, no name is displayed for the portlet, and it will be difficult to identify the portlet.



Producer Name SampleConsumer
Producer ID 10305
Available Portlets Chat
Portlet Name Chat Portlet

FIGURE 8–9 Installing Remote/WSRP Portlets

2 **Click Save.**

3 **Click on the Portlets tab to view the portlet.**

CHAPTER 9

System Maintenance

Maintaining a running implementation of WebSynergy Portal is not much different from maintaining the application server environment upon which it is running. There are, however, several factors which administrators should be aware of when they are responsible for a running instance of WebSynergy. This chapter will cover these issues, outlining for system administrators some specifics about keeping a running Liferay instance stable and secure.

The discussion on back up will cover what parts of WebSynergy should be backed up. We will not cover specific backup software or procedures; generally, most organizations have standards for doing backups of their systems, and WebSynergy as a Java EE application fits well into these standards.

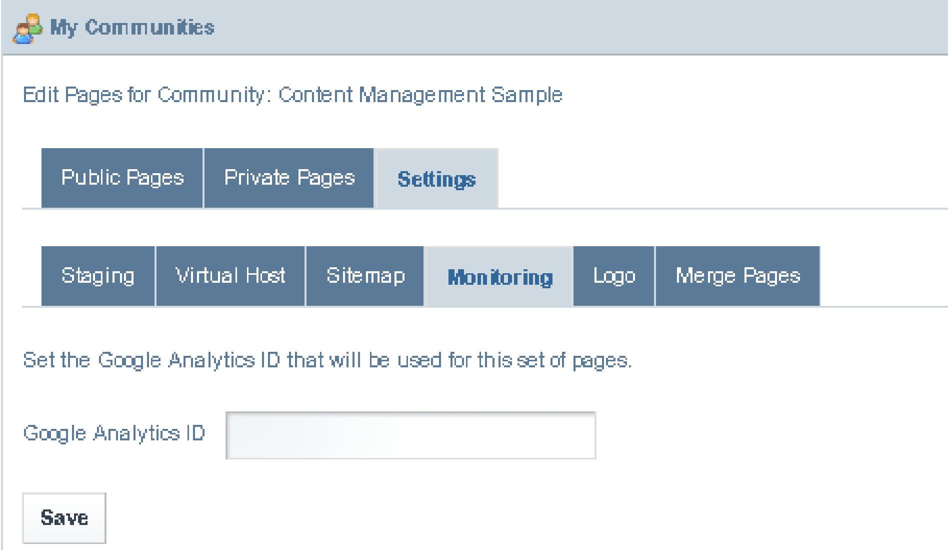
WebSynergy Monitoring using Google Analytics

WebSynergy includes built-in support for Google Analytics, allowing administrators to make use of Google's tool set for analyzing site traffic data. When you sign up for Google Analytics, a snippet of code is provided which needs to be added to your web pages in order to allow Google's system to register the page hit. It can be a tedious process to add this code to every page on a site, especially if it is a large site and there is a lot of user-generated content.

This problem can be solved in Liferay by putting Google's code into a custom theme written especially for the site on which the portal is running. Doing this, however, requires that a theme developer make specific changes to the theme, and it prevents users from using the many freely available themes that are available for WebSynergy 'out of the box'.

Because of this, support for Google Analytics has been built into WebSynergy, and can be turned on through a simple user interface. This allows WebSynergy Administrators to make use of Google Analytics on a community by community basis and turn it on and off when needed.

To enable Google Analytics support, go to the Manage Pages screen for the community for which you want to enable support. You can do this through the Communities portlet or by clicking the Manage Pages link in the Dock while you are on a page in the community. Select *Monitoring* under *Settings*.



Edit Pages for Community: Content Management Sample

Set the Google Analytics ID that will be used for this set of pages.

FIGURE 9–1 Monitoring using Google Analytics

To enable Google Analytics support, go to the Manage Pages screen for the community for which you want to enable support. You can do this through the Communities portlet or by clicking the Manage Pages link in the Dock while you are on a page in the community.

Backing Up a WebSynergy Installation

Once you have an installation of WebSynergy Portal running, you will want to have proper backup procedures in place in case of a catastrophic failure of some kind. WebSynergy is not very different from any other application that may be running in your application server, but there are some specific components that need to be backed up in addition to your regular backup procedures for your application server.

Source Code

If you have extended WebSynergy or have written portlet or theme plugins, they should be stored in a source code repository such as Subversion, CVS, or Git. This repository should be backed up on a regular basis to preserve your ongoing work.

If you are extending WebSynergy with the Extension Environment, you will want to make sure that you also store the version of the WebSynergy source on which your extension environment is based. This allows your developers convenient access to all of the tools they need to build your extension and deploy it to a server.

WebSynergy's File System

WebSynergy's configuration file, `portal-ext.properties`, gets stored in the WEB-INF/classes folder in the location to which your application server deployed WebSynergy. At a minimum, this file should be backed up, but it is generally best to back up your whole application server.

WebSynergy also stores configuration files, search indexes, cache information, and the default Jackrabbit document repository in a folder called `websynergy` in the domain directory of the application server. It is recommended to backup this folder.

Database

WebSynergy's database is the central repository for all of the Portal's information and is the most important component which needs to be backed up. You can do this by either backing up the database live (if your database allows this) or by exporting the database and then backing up the exported file. For example, MySQL ships with a `mysqldump` utility which allows you to export the entire database and data into a large SQL file. This file can then be backed up. In case of a database failure, it can be used to recreate the state of the database at the time the dump was created.

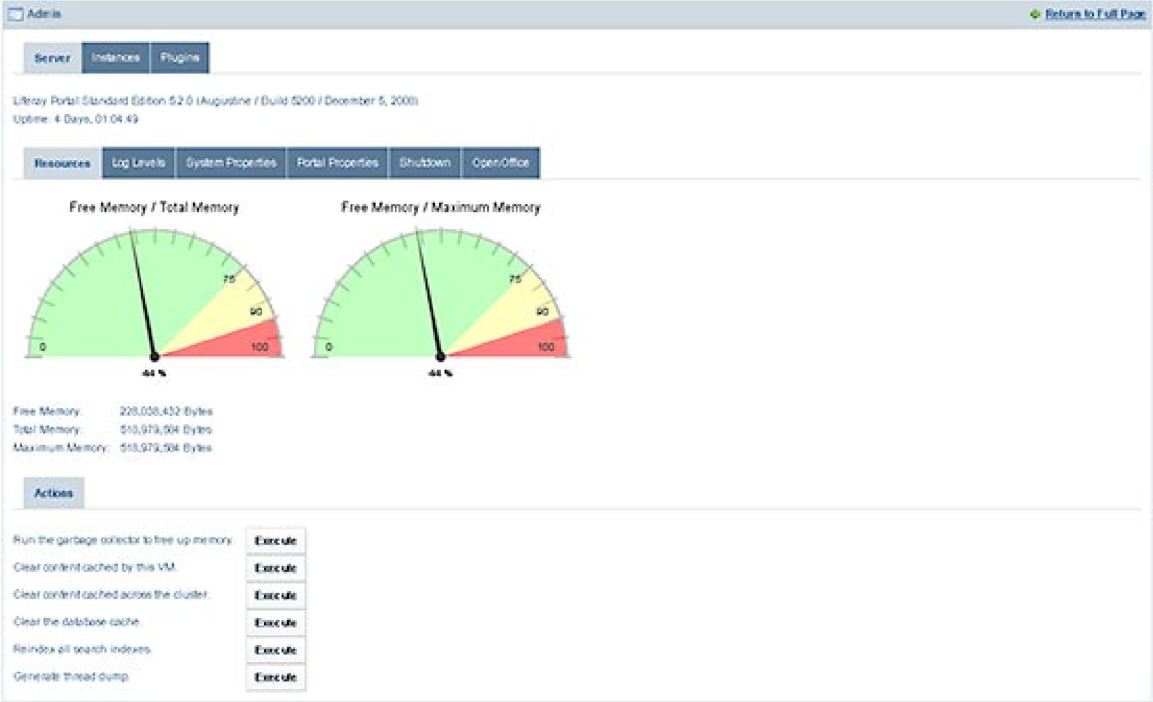
If you are using WebSynergy's Document Library extensively, it is likely that you have configured Jackrabbit to store documents in a database rather than the file system. In this case, the Jackrabbit database should be backed up also.

Monitoring and Logging

To be done.

Monitoring WebSynergy

Sign in as admin user to access the *Admin* portlet. To monitor WebSynergy, select the *Server* tab, then the *Resources* tab.



Liferay Portal Standard Edition 5.2.0 (Augustine / Build 5200 / December 5, 2008)
Uptime: 4 Days, 01:04:49

Free Memory: 228,038,432 Bytes
Total Memory: 513,979,504 Bytes
Maximum Memory: 513,979,504 Bytes

Run the garbage collector to free up memory. Execute
Clear content cached by this VM. Execute
Clear content cached across the cluster. Execute
Clear the database cache. Execute
Reindex all search indexes. Execute
Generate thread dump. Execute

FIGURE 9–2 Monitoring of resources

You can execute a resource maintenance action by selecting the *Execute* button against an action listed below *Actions*. You can execute the following actions:

Run the garbage collector to free up memory

Garbage collection frees memory occupied by unused objects. Click on the Execute button against the 'Run the garbage collector to free up memory' action. It increases the 'Free Memory' considerably.

Clear content cached by this VM

The Virtual Machine identifies the web page responses that are the same for each request

	and to remember (cache) the content. In some cases it can be useful to clear this cache to free the memory.
Clear content cached across the cluster	Executing this action clears the cluster cache.
Clear the database cache	Executing this action clears the database cache.
Reindex all search indexes	Executing this action removes all old search entries.
Generate thread dump	When the application server freezes or hangs or becomes sluggish for no apparent reason, it is recommended to generate a thread dump. Executing this action generates a thread dump.

Logging in WebSynergy

WebSynergy uses `Log4j` extensively to implement logging for nearly every class in the portal. If you need to debug something specific while a system is running, you can use the Admin portlet to set logging levels by class dynamically.

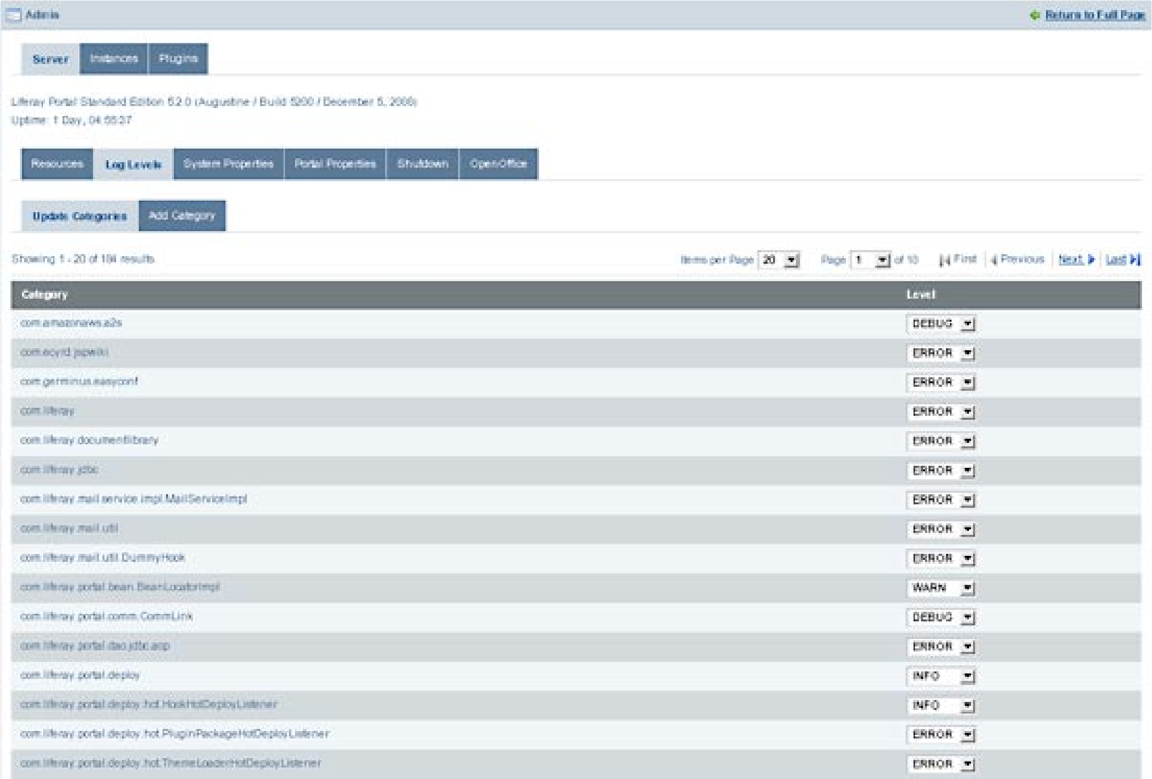
Sign in as admin user to access the *Admin* portlet. To view the log levels, select the *Server* tab, then the *Log Levels* tab.

You will then see a paginated list of logging categories. You can select a logging level for a category from the list. If the logging level is set to ERROR, an error message is displayed if an error occurs in the class. Similarly, if the logging level is set to WARN, an error message is displayed if a warning occurs in the class. OFF, FATAL, ERROR, WARN, INFO, DEBUG, and ALL are the different logging levels.

Each category is filtered by its place in the class hierarchy. For example, if you wanted to see logging for a specific class that is registered in the Admin portlet, you would browse to that specific class and change its log level to something that is more descriptive, such as DEBUG. Once you click the Save button at the bottom of the list, you will start seeing DEBUG messages from that class in your application server's log file.

If you are not sure which class you want to see log messages for, you can find a place higher up in the hierarchy and select the package name instead of an individual class name. If you do this, messages for every class lower in the hierarchy will be displayed in your application server's log file.

Be careful when you do this. If you set the log level to DEBUG somewhere near the top of the hierarchy (such as `com.liferay`, for example), you may wind up with a lot of messages in your log file. This could make it difficult to find the one you were looking for, and causes the server to do more work writing messages to its log file.



Liferay Portal Standard Edition 5.2.0 (Augustine / Build 5200 / December 5, 2008)
Uptime: 1 Day, 04:55:37

Showing 1 - 20 of 184 results

Category	Level
com.amazonaws.a2s	DEBUG
com.ecyrd.jspwiki	ERROR
com.germinus.easyconf	ERROR
com.liferay	ERROR
com.liferay.documentlibrary	ERROR
com.liferay.jdbc	ERROR
com.liferay.mail.service.impl.MailServiceImpl	ERROR
com.liferay.mail.util	ERROR
com.liferay.mail.util.DummyHook	ERROR
com.liferay.portal.bean.BeanLocatorImpl	WARN
com.liferay.portal.comm.CommLink	DEBUG
com.liferay.portal.dao.jdbc.aop	ERROR
com.liferay.portal.deploy	INFO
com.liferay.portal.deploy.hot.HookHotDeployListener	INFO
com.liferay.portal.deploy.hot.PluginPackageHotDeployListener	ERROR
com.liferay.portal.deploy.hot.ThemeLoaderHotDeployListener	ERROR

FIGURE 9–3 Changing the logging level

If you are working in the extension environment or have created a plugin and want to set the log level for one of your own classes, you can register that class (so long as it uses Log4J to do its logging) with the Admin portlet so that you can control the log levels more easily.

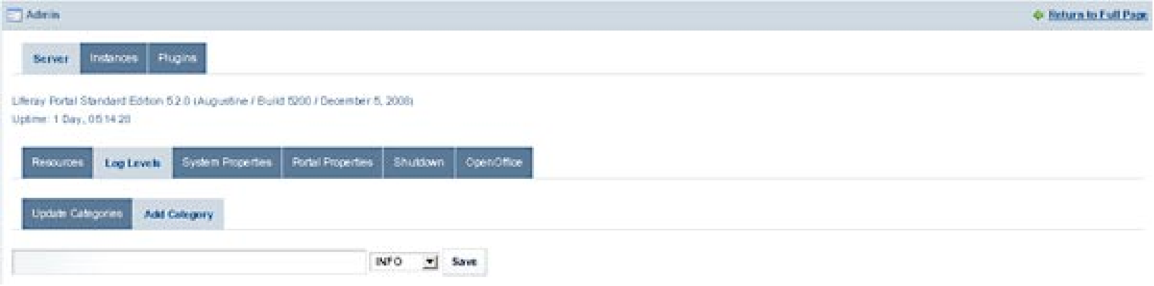
You will first need to implement Log4J logging in your class, with a statement such as the following (taken from Liferay's JCRHook class):

```
private static Log _log = LogFactory.getLog(JCRHook.class);
```

You would then use this `_log` variable to create log messages in your code for the various logging levels:

```
_log.error("Reindexing " + node.getName(), e1);
```

To enable your logging messages to appear in your server's log file via the Admin portlet, click the *Add Category* tab on the same *Log Levels* page.



Liferay Portal Standard Edition 5.2.0 (Augustine / Build 5200 / December 5, 2008)
Uptime: 1 Day, 05:14:20

FIGURE 9–4 Adding a logging category

You will see that you can add a logging category to the Admin portlet. Simply put in the fully qualified name of your class or of the package that contains the classes whose log messages you want to view, choose a log level, and then click the Save button. You will now start to see log messages from your own class or classes in the server's log file.

CHAPTER 10

Performance Tuning

To be done.

Database

To be done.

Need detailed deliberation on tuning of `my.cnf`.

Application Server

Tuning the Application Server is most important to see 'real' performance improvement.

JVM Options for Tuning the Application Server

To be done.

Thread Pools

To be done.

MySQL Database Connections

To be done.

Clustering

Clustering is needed for scalability, increased availability, and load balancing.

If you are using Sun Glassfish Enterprise Server as the application server, you can refer to the following links for the information on clustering:

- `http://developers.sun.com/appserver/reference/techart/glassfishcluster/`
- `https://glassfish.dev.java.net/javaee5/build/GlassFish_LB_Cluster.html`
- `http://wiki.glassfish.java.net/Wiki.jsp?page=GlassFishV2Architecture`
- `http://docs.sun.com/app/docs/prod/gf.entsvr.v3pre?l=en[amp ]a=view`

Memory

Memory is one of the first things to look at when you want to optimize performance. If you have any disk swapping, that will have a serious impact on performance. Make sure that your server has an optimal amount of memory and that your JVM is tuned to use it.

There are three JVM command switches that control the amount of memory it will use.

Java heap size:

`-Xms`

VM heap size:

`-Xmx`

`PermGen` space:

`-XX:MaxPermSize`

These three settings control the amount of memory available to the JVM initially, the maximum amount of memory into which the JVM can grow, and the separate area of the heap called Permanent Generation space.

For example, the default settings can be:

```
-Xms128m -Xmx1024m -XX:MaxPermSize=128m
```

This is perfectly reasonable for a moderately sized machine or a developer machine. These settings allow the JVM to initially take 128MB of RAM, grow up to 1024MB of RAM, and have a PermGen space of 128MB. If, however, you have Liferay on a server with 4GB of RAM and you are having performance problems, the first thing you might want to look at is increasing the memory available to the JVM. You will be able to tell if memory is a problem by running a profiler (such as Jprobe or YourKit) on the server. If you see Garbage Collection (GC) running frequently, you will definitely want to increase the amount of memory available to the JVM.

Issues with PermGen space can also affect performance. PermGen space contains long-lived classes, anonymous classes and interned Strings. Hibernate, in particular-which WebSynergy uses extensively? has been known to make use of PermGen space. If you increase the amount of memory available to the JVM, you may want to increase the amount of PermGen space accordingly.

Properties File Changes

There are also some changes you can make to your portal-ext.properties file once you are in a production environment.

Set the following to false to disable checking the last modified date on server side CSS and JavaScript:

```
last.modified.check=false
```

Set this property to true to load the theme's merged CSS files for faster loading for production. By default it is set to false for easier debugging for development. You can also disable fast loading by setting the URL parameter `css_fast_load` to 0.

```
theme.css.fast.load=true
```

Set this property to true to load the combined JavaScript files from the property javascript.files into one compacted file for faster loading for production. By default it is set to false for easier debugging for development. You can also disable fast loading by setting the URL parameter `js_fast_load` to 0.

```
javascript.fast.load=true
```

Also, set the following parameters:

```
portlet.css.enabled=false
```

```
velocity.engine.resource.manager.cache.enabled=true
```

```
com.liferay.portal.servlet.filters.layoutcache.LayoutCacheFilter=true
```

Servelet Filters

WebSynergy comes by default with a number of servlet filters enabled and running. It is likely that for your installation, you don't need them all. Two filters that you can disable without any impact are the `Compression Filter` and the `Strip Filter`. These filters are responsible for shrinking the size of the response (to save bandwidth). The Strip Filter removes whitespace from the response object, and the Compression Filter compresses it. This obviously requires some processing, and so disabling these two filters can enhance performance.

To disable a servlet filter, simply comment it out of your `web.xml` file.

If there is a feature supported by a servlet filter that you know you are not using, you can comment it out as well to achieve some performance gains. For example, if you are not using CAS for single sign-on, comment out the `CAS Filter`. If you are not using NTLM for single sign-ons, comment out the `Ntlm Filter`. If you are not using the Virtual Hosting for Communities feature, comment out the `Virtual Host Filter`. The fewer servlet filters you are running, the less processing power is needed for each request.

Portlets

WebSynergy comes pre-bundled with many portlets which contain a lot of functionality, but not every web site that is running on WebSynergy needs to use them all. In `portlet.xml` and `liferay-portlet.xml`, comment out the ones you are not using. While having a loan calculator, analog clock, or game of hangman available for your users to add to pages is nice, those portlets may be taking up resources that are needed by custom portlets you have written for your site. If you are having performance problems, commenting out some of the unused portlets may give you the performance boost you need.

Java Environment

To be done.

Portal Extensions

To be done.

CHAPTER 11

Troubleshooting

This chapter contains troubleshooting information for some of the issue that you may face when you are working with WebSynergy.

Troubleshooting for the Issues Related with Installation

If you are deploying WebSynergy on an existing installation of GlassFish, you need to determine the install location of GlassFish. If GlassFish is installed at the root directory, you can deploy WebSynergy either at the root directory or at the user directory. If GlassFish is installed at your user directory, you must install WebSynergy at the same user directory.

In the scenario involving JDK, GlassFish, and WebSynergy; WebSynergy is at the top layer, GlassFish is in the middle layer, and JDK is in the bottom layer. When you are using the GlassFish WebSynergy bundle, you need to determine the install location of JDK. If JDK is at the root directory, you can install the bundle either at the root directory or at your user directory. If JDK is installed at the user directory, you need to install the bundle at the same directory.

Note – It is a good practice to have your GlassFish installation dedicated only for running WebSynergy.

Suppose you have JDK 1.5 installed in the root directly and you have installed JDK 1.6 in your user directory. Because WebSynergy often works better on a more recent JDK version, you might want to override the installation at the root directory.

▼ To Reset the `JAVA_HOME` Variable to the User Directory

1. **Type the following commands in a terminal window:**

   ```
   export JAVA_HOME JDK-install-dir
   export PATH=$JAVA_HOME/bin:$PATH
   ```

2. **To verify the change, type the following command:**

   ```
   echo $JAVA_HOME
   ```

   The new directory should be listed.

▼ To Set the `JAVA_HOME` Environment Variable to the User Directory on Non Solaris Operating Systems

1. **Open the *GlassFish-install-dir*`/config/asenv.conf/asenv.bat` file.**
2. **Set the `AS_JAVA` variable to the JDK 1.6 user installation.**
3. **Reinstall the WebSynergy bundle and start the domain.**

Troubleshooting for Issues Related with Accessing the HTTP Instance

You can access the HTTP instance for the WebSynergy installation on your machine at `http://localhost:8080/`. In case you are installing WebSynergy on a remote machine, you access the HTTP instance at `http://`<*machine-name*>`:8080/`.

Troubleshooting for Issues Related with User Activities

Users may face some issues when they are working with WebSynergy. Workaround for those issues are discussed in this section.

▼ To Create a Private Page

If your user account has only public pages and you want to add a private page to it, follow the below procedure.

1. **Choose Control Panel from the welcome dock.**

2 **Click on My Pages under My Account.**

3 **Click on the Private Pages tab.**

4 **Specify a name for the page and click on the Add Page button.**

Troubleshooting for Issues Related with Installing Plugins

The property called `auto.deploy.dest.dir` is essential for installing plugins. If you are failing to install plugins, the reason can be that the property is not defined properly. GlassFish defines the hot deploy folder as a folder called *autodeploy* inside of the domain folder in which your server is running. By default, this is in <*Glassfish Home*>`/domains/domain1/autodeploy`. JBoss defines the hot deploy folder as a root folder inside of the particular server configuration you are using. By default, this is in <*JBoss Home*>`/server/default/deploy`. WebLogic defines this folder inside of the domain directory. By default, this is in <*Bea Home*>`/user_projects/domains/<domain name>/ autodeploy`.

You will first need to determine where the hot deploy folder is for the container you are running. Consult your product documentation for this. Once you have this value, there are two places in which you can set it: the `portal-ext.properties` file and in the *Plugin Installer* portlet.

To change this setting in the `portal-ext.properties` file, browse to where WebSynergy was deployed in your application server. Inside of this folder should be a `WEB-INF/classes` folder. Here you will find the `portal-ext.properties file`. Open this file in a text editor and look for the property `auto.deploy.dest.dir`. If it does not appear in the file, you can add it. The safest way to set this property is to define the property using an absolute path from the root of your file system to your application server's hot deploy folder. For example, if you are using GlassFish, and you have the server installed in `/java/glassfish`, your `auto.deploy.dest.dir` property would look like the following:

```
auto.deploy.dest.dir=/java/glassfish/domains/domain1/autodeploy
```

Remember, if you are on a Windows system, use forward slashes instead of back slashes, like so:

```
auto.deploy.dest.dir=C:/java/glassfish/domains/domain1/autodeploy
```

Save the file and then restart your container. Now plugins should install correctly.

Common Questions

The following are some commonly asked questions about WebSynergy software.

Question: What databases does WebSynergy support?

Answer: HSQL, MySQL, Microsoft SQL, and Oracle 10g.

Question: What is the recommended JDK version to use with WebSynergy?

Answer: JDK 1.6.0_07 later. Also make sure that Ant 1.7.0 is installed on the machine on which you are installing WebSynergy.

Question: What can I do if the *Compose* link does not working in the Mail portlet?

Answer: Click the compose window twice to activate the window.

Question: What size image should I use for my site logo?

Answer: It is recommended that your site logo be no larger than 50 x 50 pixels.

Question: What is BSOJ?

Answer: BSOJ stands for big screen of junk (text characters). You may sometimes encounter this when some error has occurred in the WebSynergy user interface. If you encounter a BSOJ, the workaround is to log out of WebSynergy, close your browser, then reopen the browser and log in again.

Question: What is the recommended amount of memory (RAM) to use with WebSynergy?

Answer: It is recommended that the machine on which a WebSynergy server is running has a minimum of 2GB RAM. Client machines connecting to a WebSynergy site should have a minimum of 1GB RAM.

Question: What Web browser is recommended for use with WebSynergy?

Answer: Firefox 2.*x* or 3.*x* are recommended and are the test bases on which WebSynergy has been most extensively tested. Microsoft Internet Explorer is know to have some incompatibilities with WebSynergy.

Question: On what port does WebSynergy run?

Answer: By default, WebSynergy runs on port `8080`. The port can be changed later, but the WebSynergy installation program runs on port `8080`, so this port must be free before you start the installation process. If upgrading from a previous WebSynergy installation, be sure to stop any running GlassFish domains before proceeding.

Question: How to create a custom portlet bundle?

Answer: You can use *Add Application* to add portlets and widgets to a page. You can add portlets and widgets to the public pages and private pages of Communities and Organizations and propagate those Communities and organizations to production environment. the portlets and widgets thus bundled with a Community or an Organization can be considered as custom portlet bundles.