

# JBoss Portal 2.0 Final

## User Guide

---

# Table of Contents

JBoss Portal - Overview .....	iv
Feature List .....	v
Target Audience .....	viii
Acknowledgements .....	ix
1. Supported versions .....	1
1.1. JBoss application server .....	1
1.2. Database .....	1
1.3. Source building .....	2
2. Installation .....	3
2.1. Install Overview .....	3
2.1.1. JBoss Application Server .....	3
2.1.2. Getting the JBoss Portal files .....	3
2.1.3. Database .....	3
2.1.4. JDBC connector .....	4
2.2. Setting up your environment .....	4
2.2.1. Database .....	4
2.2.2. JBoss AS Configuration .....	4
2.3. Deploying JBoss Portal .....	4
2.3.1. From the binaries .....	5
2.3.2. From the sources .....	5
2.4. Running JBoss Portal .....	6
2.5. Customizing your installation .....	6
2.5.1. Changing the port .....	6
2.5.2. Changing the context path .....	6
2.5.3. Rebuilding the DB tables .....	6
2.5.4. Forcing the DB dialect .....	7
3. User Portlet .....	8
3.1. Introduction .....	8
3.2. Functionalities .....	8
3.3. Configuration .....	8
4. Role Portlet .....	11
4.1. Introduction .....	11
4.2. Functionalities .....	11
4.2.1. Create a role .....	11
4.2.2. Edit a role .....	11
4.2.3. Editing Role Members .....	11
5. CMS Portlet .....	12
5.1. Introduction .....	12
5.2. Functionalities .....	12
5.3. Portlet Configuration .....	12
5.3.1. RDBMS Configuration .....	13
6. AdminCMS Portlet .....	15
6.1. Introduction .....	15
6.2. Functionalities .....	17

6.2.1. Viewing Directories .....	17
6.2.2. Viewing File Properties .....	17
6.2.3. Copying Files/Directories .....	18
6.2.4. Moving Files/Directories .....	18
6.2.5. Deleting Files/Directories .....	19
6.2.6. Creating Directories .....	19
6.2.7. Creating Text/HTML Files .....	19
6.2.8. Uploading Binary Files .....	21
6.2.9. Editing Text/HTML Files .....	21
7. Permissions Portlet .....	23
7.1. Introduction .....	23
7.2. Functionalities .....	23
7.2.1. Add a role to a permission .....	23
8. Forums Portlet .....	27
8.1. Introduction .....	27
8.2. Functionalities .....	28
8.3. Installation .....	29
8.4. Configuration .....	29
8.5. Security .....	29

---

# JBoss Portal - Overview

Many IT organizations look to achieve a competitive advantage for the enterprise by improving business productivity and reducing costs. Today's top enterprises are realizing this goal by deploying enterprise portals within their IT infrastructure. Enterprise portals simplify access to information by providing a single source of interaction with corporate information. Although today's packaged portal frameworks help enterprises launch portals more quickly, only JBoss Portal can deliver the benefits of a zero-cost open source license combined with a flexible and scalable underlying platform.

JBoss Portal 2.0 provides an open source and standards-based environment for hosting and serving a portal's Web interface, publishing and managing its content, and customizing its experience. It is entirely standards-based and supports the JSR-168 portlet specification, which allows you to easily plug-in standards-compliant portlets to meet your specific portal needs. JBoss Portal 2.0 is available through the business-friendly LGPL [<http://www.jboss.com/company/aboutopensource>] open source license and is supported by JBoss Inc. Professional Support and Consulting [<http://www.jboss.com/services/index>]. JBoss support services are available to assist you in designing, developing, deploying, and ultimately managing your portal environment. JBoss Portal is currently developed by JBoss, Inc. developers, Novell developers, and community contributors.

The JBoss Portal 2.0 framework and architecture includes the portal container and supports a wide range of features including standard portlets, single sign-on, clustering and internationalization. Portal themes and layouts are configurable. Fine-grained security administration down to portlet permissions rounds out the security model. JBoss Portal 2.0 includes a rich content management system and message board support.

## JBoss Portal Resources:

1. JBoss Portal Home Page [<http://www.jboss.org/products/jbossportal>]
2. Forums: User [<http://www.jboss.org/index.html?module=bb&op=viewforum&f=215>] | Developer [<http://www.jboss.org/index.html?module=bb&op=viewforum&f=205>]
3. Wiki [<http://www.jboss.com/wiki/Wiki.jsp?page=JBossPortal>]
4. Our Roadmap  
[<http://jira.jboss.com/jira/browse/JBPORTAL?report=com.atlassian.jira.plugin.system.project:roadmap-panel>]

The JBoss Portal team encourages you to use this guide to install and configure JBoss Portal. If you encounter any configuration issues or simply want to take part in our community, we would love to hear from you in our forums.

---

# Feature List

The following list details features found in this document's related release. Currently, this is JBoss Portal 2.0 Final. For a technical view of our features, view the Project Roadmap and Task List [<http://jira.jboss.com/jira/browse/JPORAL>].

## Technology and Architecture

- **JEMS:** Leverages the power of JBoss Enterprise Middleware Services : JBoss Application Server, JBoss Cache, JGroups, and Hibernate.
- **DB Agnostic:** Will work with any RDBMS supported by Hibernate
- **SSO/LDAP:** Leverages Tomcat and JBoss single sign on (SSO) solutions.
- **JAAS Authentication:** Custom authentication via JAAS login modules.
- **Cacheing:** Utilizes render-view caching for improved performance.
- **Clusterable:** Cluster support allows for portal state to be clustered for all portal instances.
- **Hot-Deployment:** Leverages JBoss dynamic auto deployment features.
- **SAR Installer:** Browser-based installer makes installation and initial configuration a breeze.

## Portal and Portal Container

- **Multiple Portal Instances:** Ability to have multiple Portal instances running inside of one Portal container.
- **Internationalization:** Ability to use internationalization resource files for every portlet.
- **Pluggable services:** Authentication performed by the servlet container and JAAS make it possible to swap the authentication scheme.
- **Page-based Architecture:** Allows for the grouping/division of portlets on a per-page basis.
- **Existing Framework support:** Portlets utilizing Struts, Spring MVC, Sun JSF-RI, or MyFaces are supported.

## Themes and Layouts

- **Easily swappable themes/layouts:** New themes and layouts containing images can be deployed in WAR archives.
- **Flexible API:** Theme and Layout API are designed to separate the business layer from the presentation layer.
- **Per-page layout strategy:** Different layouts can be assigned to different pages.

## User and Group Functionality

- **User registration/validation:** Configurable registration parameters allow for user email validation before activation.
- **User login:** Makes use of servlet container authentication.
- **Create/Edit Users:** Ability for administrators to create/edit user profiles.
- **Create/Edit Roles:** Ability for administrators create/edit roles.
- **Role Assignment:** Ability for administrators to assign users to roles.

### Permissions Management

- **Extendable permissions API:** Allows custom portlets fine-grained permissions based on role definition or portal context.
- **Administrative interface:** Allows for fine-grained permissions assignments to roles at any time for any deployed portlet.

### Content Management System

- **Full WebDAV support:** CMS store implements Jakarta Slide [<http://jakarta.apache.org/slide/index.html>] WebDAV system. Allows for native OS access to file store.
- **DB or Filesystem store support:** Configurable content store to either a filesystem or RDBMS.
- **Versioning support:** All content edited/created is autoversioned with a history of edits that can be viewed at any time.
- **Content Serving Search-engine-friendly URLs:** <http://yourdomain/portal/default/index.html> (Does not apply to portlet actions.)
- **No long portal URLs:** Serve binaries with simple urls. (<http://domain/files/products.pdf>)
- **Bundled HTML Portlet:** Allows for extra instances of static content from the CMS to be served under separate windows.
- **Directory Support:** create, move, delete, and copy entire directory trees.
- **File Functions:** create, move, copy, and delete files.
- **Custom error pages:** Error pages can be edited to suit the needs of your website - 404, invalid login pages.
- **Embedded directory-browser:** When copying, moving, deleting, or creating files, administrators can simply navigate the directory tree to find the collection they want to perform the action on.
- **Ease-of-use architecture:** All actions to be performed on files and folder are one mouse-click away.
- **Full-featured HTML editor:** HTML Editor contains WYWIWYG mode, preview functionality, and HTML source editing mode. HTML commands support tables, fonts, zooming, image and url linking, flash movie support, bulleted and numbered list, and dozens more.

- **Editor style-sheet support:** WYSIWYG editor displays current Portal style-sheet, for easy choosing of classes.

### Message Boards

- **Instant reply:** Instant reply feature, makes for one-click replies to posts.
- **Post quoting:** Quote an existing topic and poster within a reply.
- **Flood control:** Prevents abuse of multiple posts withing a set configurable time-frame.
- **Category creation:** Create a category that contains forums within it.
- **Forum creation:** Create a forum and assign it to a specific category.
- **Forum modification:** Edit, move, delete forums.
- **Forum and category reordering:** Reorder categories and forums in the order you would like them to appear on the page.

---

# Target Audience

This document is intended for those using JBoss Portal. It details the default features found within the standard Portal distribution and addresses configuration issues found in each component. For developers wanting to develop and deploy custom portlets, create/deploy custom themes, or utilize the JBoss Portal API, there is a reference document available.



---

# Acknowledgements

We would like to thank **all** the developers that participate in the JBoss Portal project effort.

Specifically,

1. Remy for his help with Tomcat configuration.
2. The Nodesk team that gave us our default theme.
3. Kev "kevs3d" Roast for supplying us with two working portlets that integrate existing frameworks in to the portal: Sun JSF-RI and Spring MVC Portlet. These formed the base for our sample pack.
4. Swarn "sdhaliwal" Dhaliwal for supplying us with the Struts-Bridge, that will allow for existing struts applications to work with the Portal.

Contributions of any kind are always welcome, you can contribute by providing ideas, filling bug reports, producing some code, designing a theme, writing some documentation, etc... To report a bug please use our Jira system [<http://jira.jboss.com>].

## Supported versions

**Thomas Heute** <theute@jboss.org>

**Roy Russo** <roy@jboss.org>

Here is a list of tested versions or reported as working by users, before reporting a problem please make sure that you are using a compatible version.

If you successfully installed JBoss portal on versions not listed here please let us know so we can add it here.

### 1.1. JBoss application server

#### Note

Currently we recommend ONLY using JBoss AS 4.0.2, or greater. Previous versions of JBoss Application Server are not supported, as JBoss Portal is configured to work with Hibernate3.

### 1.2. Database

#### Note

JBoss Portal employs Hibernate as an interface to RDBMS. Any DB supported by Hibernate will work with JBoss Portal.

MySQL is the database used for our testings, 4.0.22 is one version that has been fully tested.

- MySQL 4.0 (4.0.22 is known to be supported along with the connector 3.0.16)
- MySQL 4.1 (4.1.7 is known to be supported along with the connector 3.0.16)
- MySQL 4.1 (4.1.8 is known to be supported along with the connector 3.1)
- PostgreSQL (reported to work)
- HypersonicSQL (works)
- Oracle (reported to work)
- MSSQL (not tested)

## 1.3. Source building

The source building mechanism works on Windows, Linux, MacOS X and any 'Unix like' operating system.

## Installation

**Thomas Heute** <theute@jboss.org>

**Roy Russo** <roy at jboss dot org>

### 2.1. Install Overview

There are a couple of archives that you will need to download in order to install JBoss Portal.

#### 2.1.1. JBoss Application Server

Of course you will need to install JBoss Application Server prior to install JBoss portal, if you didn't do so yet, please install JBoss 4.0.2 from Sourceforge [[http://sourceforge.net/project/showfiles.php?group\\_id=22866&package\\_id=16942&release\\_id=312621](http://sourceforge.net/project/showfiles.php?group_id=22866&package_id=16942&release_id=312621)].

#### 2.1.2. Getting the JBoss Portal files

You can download JBoss portal in different ways, packaged in binaries, sources or from the CVS.

- Packaged: From the JBoss portal project page [<http://www.jboss.org/products/jbossportal#files>]
- CVS Version 2.0 RC2: `cvs -d :pserver:anonymous@anoncvs.forge.jboss.com:/cvsroot/jboss co -r JBoss_Portal_2_0_0 jboss-portal-2.0`
- CVS Version HEAD (The most up to date sources at your own risks): `cvs -d :pserver:anonymous@anoncvs.forge.jboss.com co jboss-portal-2.0`

#### Warning

Do not attempt to get the jboss-portal module. The latest release *is* jboss-portal-2.0

#### 2.1.3. Database

You will need a database to store the data of the system, you can use any database supported by Hibernate. We have tested JBoss Portal on the following, but other should work just the same:

- MySQLDB

- Hypersonic DB
- PostGreSQL
- Oracle 10g

### 2.1.4. JDBC connector

You must make sure that your JDBC connector for your database is under `server/default/lib`. The MySQL JDBC connector is available here [<http://dev.mysql.com/downloads/connector/j/3.1.html>], the postgresSQL JDBC connector is available here [<http://jdbc.postgresql.org/download.html>].

## 2.2. Setting up your environment

### 2.2.1. Database

All databases supported by hibernate are support by JBoss Portal. Below is a generic ordered list of steps that should be followed on any DB:

1. Create a new Database. For MySQL we name it *jbossportal*.
2. Give access rights to whatever user with whatever password to this new database. For MySQL we create a user "*portal*" and give him a password "*portalpassword*", and grant him rights to the jbossportal DB.

#### Note

As of JBoss Portal RC2, all database tables will be created for you at runtime. The only thing you need to make certain is that there is a database created, a working JDBC connector, and that the user/password combination works.

### 2.2.2. JBoss AS Configuration

If you need a custom setup of JBoss AS, you should read the documentation about JBoss application server. In our case, we will use the default configuration shipped with JBoss AS 4.0.2.

At this stage you should have the `jboss-4.0.2.zip` or any other archive of the same version. First you need to setup `JBOSS_HOME` environment variable otherwise you won't be able to compile JBoss Portal. To do so go to `Start > Settings > Control Panel > System > Advanced > Environment Variables`, and add the `JBOSS_HOME` environment variable. Or do `export JBOSS_HOME=/path/to/your/jboss/directory` on a Unix-like system.

## 2.3. Deploying JBoss Portal

### 2.3.1. From the binaries

The downloaded archive contains the following files:

- jboss-portal.sar
- portal-forums.ear
- /setup/portal-\*-ds.xml

#### Note

It is important that you configure the correct datasource file under /setup. There are a few already created for support of popular databases. You can also create your own. Please verify that the username, password, url, and driver-class are correct for your flavor of DB. You can deploy the datasource file by itself to test, in advance.

1. Copy/Move jboss-portal.sar, portal-forums.ear, and your configured portal datasource file to \$JBOSS\_HOME/server/default/deploy

### 2.3.2. From the sources

1. First, build the sources and deploy them, go to jboss-portal-2.0/build and type `sh build.sh deploy`, you should read `BUILD SUCCESSFUL` at the end of the operation. This operation should have copied the jboss-portal.sar to your \$JBOSS\_HOME/server/default/deploy directory.

#### Warning

Make sure that JBOSS\_HOME is still defined in the environment or it will not work.

2. Now you will need to build the datasource files for DB access. To do so go to jboss-portal-2.0/core and type `sh build.sh datasource`. It will create all the files under jboss-portal-2.0\core\output\resources\setup.

#### Note

It is important that you configure the correct datasource file jboss-portal-2.0\core\output\resources\setup. There are a few already created for support of popular databases. You can also create your own. Please verify that the username, password, url, and driver-class are correct for your flavor of DB. You can deploy the datasource file by itself to test, in advance.

3. Before you deploy the application by itself, you will need to have the database deployment descriptor (portal-\*-ds.xml) in the \$JBOSS\_HOME\server\default\deploy directory. To do so copy the correct portal-\*-ds.xml file in to the /deploy directory.
4. You will also need to put the jar file of your database connector in \$JBOSS\_HOME\server\default\lib, if you have not already done so.

## 2.4. Running JBoss Portal

Now you can start JBoss AS by going into `$JBOSS_HOME/bin` and typing `run`. All database tables, cms directories, and initial content for each will be created/inserted during the startup process, if it does not exist.

### Warning

If the forums portlet is not installed, the link to the forums page will not work, see the forums portlet chapter of this document for the installation instructions.

Using your browser, navigate to `http://localhost:8080/portal` and you should see the portal.

## 2.5. Customizing your installation

### 2.5.1. Changing the port

It is common to have a server running on the port 80 instead of the default port 8080, to change that you will need to change it at two levels:

- To change it at the server level, you need to edit the file `$JBOSS_HOME/server/default/deploy/jbossweb-tomcat50.sar/server.xml` and change the port value of the HTTP Connector to the same value as you defined in the portal.
- Next, you will need to modify the `web.xml` descriptor in the CMSPortlet for it to be able to find the webDAV server. You can find more information about how to do this [here](#).

### Note

You can also change in those same files, the value of the SSL port, by default it is set to 8443

Now you can restart JBoss and use the new port that you defined. On systems like Linux, you need privileges to be able to run a server on a port lower than 1000, starting JBoss on the port 80 as a regular user will not work, for testing you can log as root but is not recommended if the server is public as it could be a security breach in your system.

### 2.5.2. Changing the context path

By default, the "main" page of JBoss portal will be accessible at `http://localhost:8080/portal/index.html`. You may want to change that either to a different name or to `http://localhost:8080/index.html`.

### Note

To do so, edit the file `$PORTAL_HOME/build/local.properties` and change `portal.context-root` to anything you want.

Now you can rebuild JBoss portal and redeploy it for the context path changes to take effect.

### 2.5.3. Rebuilding the DB tables

If you would like to re-create the initial database tables, ie. start with a blank install, you can do it by following these steps:

1. Log in to the JMX-Console: <http://localhost:8080/jmx-console/>
2. Locate the portal hibernate mbean [<http://localhost:8080/jmx-console/HtmlAdaptor?action=inspectMBean&name=portal%3Aservice%3DHibernate>]. From this screen, you can execute operations.
3. To drop the database tables and recreate them, first execute `destroySchema()`. Then execute, `createSchema()`. Finally execute `createContent()`.

The database should now be restored to how it was on initial install and deployment.

## 2.5.4. Forcing the DB dialect

If you encounter that the Hibernate dialect is not working properly and would like to override the default behaviour, you simply need to modify `jboss-portal.sar/conf/hibernate/hibernate.cfg.xml`. A list of supported dialects for Hibernate3, can be found here [[http://www.hibernate.org/hib\\_docs/v3/reference/en/html/session-configuration.html#configuration-optional-dialects](http://www.hibernate.org/hib_docs/v3/reference/en/html/session-configuration.html#configuration-optional-dialects)].

```
<?xml version='1.0' encoding='utf-8'?>
<!DOCTYPE hibernate-configuration PUBLIC
    "-//Hibernate/Hibernate Configuration DTD//EN"
    "http://hibernate.sourceforge.net/hibernate-configuration-3.0.dtd">
<hibernate-configuration>
  <session-factory
    name="java:portal/SessionFactory">
    <property name="connection.datasource">java:@portal.datasource.name@</property>
    <property name="show_sql">false</property>

    <!-- Force the dialect instead of using autodetection -->
    <!--
    <property name="dialect">org.hibernate.dialect.PostgreSQLDialect</property>
    -->

    <!-- Mapping files -->
    <mapping resource="org/jboss/portal/core/impl/role/RoleImpl.hbm.xml"/>
    <mapping resource="org/jboss/portal/core/impl/user/UserImpl.hbm.xml"/>
    <mapping resource="org/jboss/portal/core/impl/user/UserPref.hbm.xml"/>
    <mapping resource="org/jboss/portal/core/impl/user/UserPrefSet.hbm.xml"/>
  </session-factory>
</hibernate-configuration>
```



## User Portlet

Thomas Heute <theute@jboss.org>

### 3.1. Introduction

The user portlet is dedicated to create and manage users and their profiles.

### 3.2. Functionalities

Managing users using the user module consists in:

1. Allowing a user to register a new profile with a login and password
2. Allowing a user to log in the system and be recognized as member of user groups
3. Allowing a user to update a profile with more information like his real name, instant messenger informations...
4. Allowing an administrator to see the list of users, from there he can assign roles to a user
5. Allowing as administrator to assign roles to a user
6. Allowing a user to logout

### 3.3. Configuration

The following xml block is the standard configuration for the UserPortlet found in *portal-core.war/WEB-INF/portlet.xml*:

```
<portlet>
  <portlet-name>UserPortlet</portlet-name>
  <portlet-class>org.jboss.portal.core.portlet.user.UserPortlet</portlet-class>
  <supported-locale>en</supported-locale>
  <supported-locale>fr</supported-locale>
  <resource-bundle>Resource</resource-bundle>
  <supports>
    <mime-type>text/html</mime-type>
    <portlet-mode>VIEW</portlet-mode>
  </supports>
  <portlet-info>
    <title>User portlet</title>
  </portlet-info>
  <init-param>
```

```
<description>Whether we should use ssl on login and throughout the Portal. 1=yes;0=no</description>
<name>useSSL</name>
<value>0</value>
</init-param>
<init-param>
<description>Subscription mode</description>
<name>subscriptionMode</name>
<!--      <value>emailVerification</value>-->
<value>automatic</value>
</init-param>
<init-param>
<description>Domain of your website for email verification.</description>
<name>emailDomain</name>
<value>JBoss.com</value>
</init-param>
<init-param>
<description>Email displayed in the TO field</description>
<name>emailFrom</name>
<value>jbossportal@example.com</value>
</init-param>
<init-param>
<description>Default role of registered users</description>
<name>defaultRole</name>
<value>Users</value>
</init-param>
</portlet>
```

The following attributes can be modified in the xml descriptor:

1. useSSL

Allows for user logins to be passed thru a SSL.

a. 0

Set to zero to disable.

b. 1

Set to 1 to enable. You must have SSL configured properly in tomcat for this to work.

2. subscriptionMode

a. automatic

The user can register and is automatically enabled

b. emailVerification

The user is disabled until he clicks on a link sent to his email address.

3. emailDomain

Your domain name or the name of your website for the email verification form text.

4. emailFrom

Email address that will appear in the "From" header when the email verification is sent.

5. `defaultRole`

Default role assigned to new users

## Role Portlet

Thomas Heute <theute@jboss.org>

Roy Russo <roy@jboss.org>

### 4.1. Introduction

The role portlet is dedicated to create and edit roles. A role will be used to grant different permission level to different portlets. A user can have several roles (for example he can be an administrator of a category of forum but only a reader on another category)



### 4.2. Functionalities

#### 4.2.1. Create a role

To create a new role, you just need to define a short name that will be used for reference, and a display name for displaying to the user, for example `admin` would be a good name for the display name `Administrators`, changing the display name will not affect the security rules

#### 4.2.2. Edit a role

While editing a role, you just need to pick an existing role then change the display name.

#### 4.2.3. Editing Role Members

Allows for an administrator to search and modify the members' assigned roles.

## CMS Portlet

**Roy Russo** <roy @ jboss dot org>

### 5.1. Introduction

The CMS Portlet displays content from the file store inside a portlet window, or, in the case of binary content, outside of the portlet window altogether.

#### Note

All of your content resides under `JBOSS_HOME/server/standard/data/portal/`, if an administrator used the default filesystem store settings JBoss Portal ships with. You should not modify any of the directories/files under there, as it may cause the CMS portlet to not function properly.

### 5.2. Functionalities

As of JBoss Portal Beta 1, the CMSPortlet handles all requests for all content types.

The methodology of serving content within the CMSPortlet, allows for some beneficial features, like:

1. Search-engine friendly URLs: `http://domain/[portal]/company.html`
2. No need to modify existing web page links. links to resources (images, web pages, pdfs, zips) can all be maintained as relative.
3. Serve binaries with simple urls independant of the portal: `http://domain/products.pdf`

### 5.3. Portlet Configuration

JBoss Portal uses Jakarta Slide as its WebDAV repository. This allows most web and file browsers to navigate to a shared repository and, given the proper rights, allows for modification of content.

Here is the default configuration for the CMS respository found under `portal-core.war/WEB-INF/web.xml`:

```
<?xml version="1.0"?>
  <!DOCTYPE web-app PUBLIC
    "-//Sun Microsystems, Inc.//DTD Web Application 2.3//EN"
    "http://java.sun.com/dtd/web-app_2_3.dtd">
    <web-app>
      <context-param>
```

```

<param-name>URL</param-name>
<param-value>http://localhost:8080/webdav/files</param-value>
<!-- Comment out for SSL communication between Portal and Webdav -->
<!-- <param-value>http://localhost:443/webdav/files</param-value> -->
<description>WebDAV server URL</description>
</context-param>
<context-param>
<param-name>rootdir</param-name>
<param-value>/files</param-value>
<description>WebDAV Directory Root</description>
</context-param>
</web-app>

```

1. **URL:** This is the absolute URL for the webdav store. Comment out the appropriate *param-value* for SSL communication between the portal and webdav layer.
2. **rootdir:** specifies the base root directory for the WebDAV store. Under normal circumstances this should not be changed.

### 5.3.1. RDBMS Configuration

One additional item that is worth noting is that the slide store can be configured to access from a RDBMS using the Portal DataSource instead of the default filesystem store, by modifying portal-cms.sar/META-INF/jboss-service.xml.

```

<!-- The slide configuration using the database. Uncomment this and comment the above tx store for Database
<!--
<definition>
<store name="j2ee">
<nodestore classname="org.apache.slide.store.impl.rdbms.J2EEStore">
<parameter name="datasource">@portal.datasource.name@</parameter>
<parameter name="adapter">org.apache.slide.store.impl.rdbms.MySqlRDBMSAdapter</parameter>
<parameter name="compress">false</parameter>
</nodestore>
<securitystore>
<reference store="nodestore"/>
</securitystore>
<lockstore>
<reference store="nodestore"/>
</lockstore>
<revisiondescriptorsstore>
<reference store="nodestore"/>
</revisiondescriptorsstore>
<revisiondescriptorstore>
<reference store="nodestore"/>
</revisiondescriptorstore>
<contentstore>
<reference store="nodestore"/>
</contentstore>
</store>
<scope match="/" store="j2ee"/>
</definition>
-->

```

To activate the DB store, uncomment the above configuration and comment the below information:

```

<!-- The slide configuration using the file system. -->
<definition>
...

```

```
</definition>
```

To create the proper DB tables, you will need to execute one the schema files found in this archive [<http://download.jboss.org/jbossportal/slide/slide-schema.zip>]

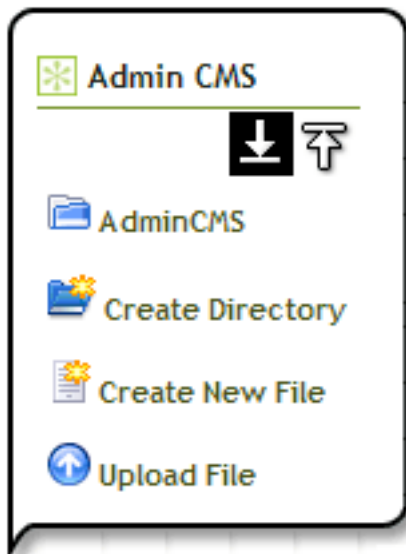
## AdminCMS Portlet

Roy Russo <roy @ jboss dot org>

### 6.1. Introduction

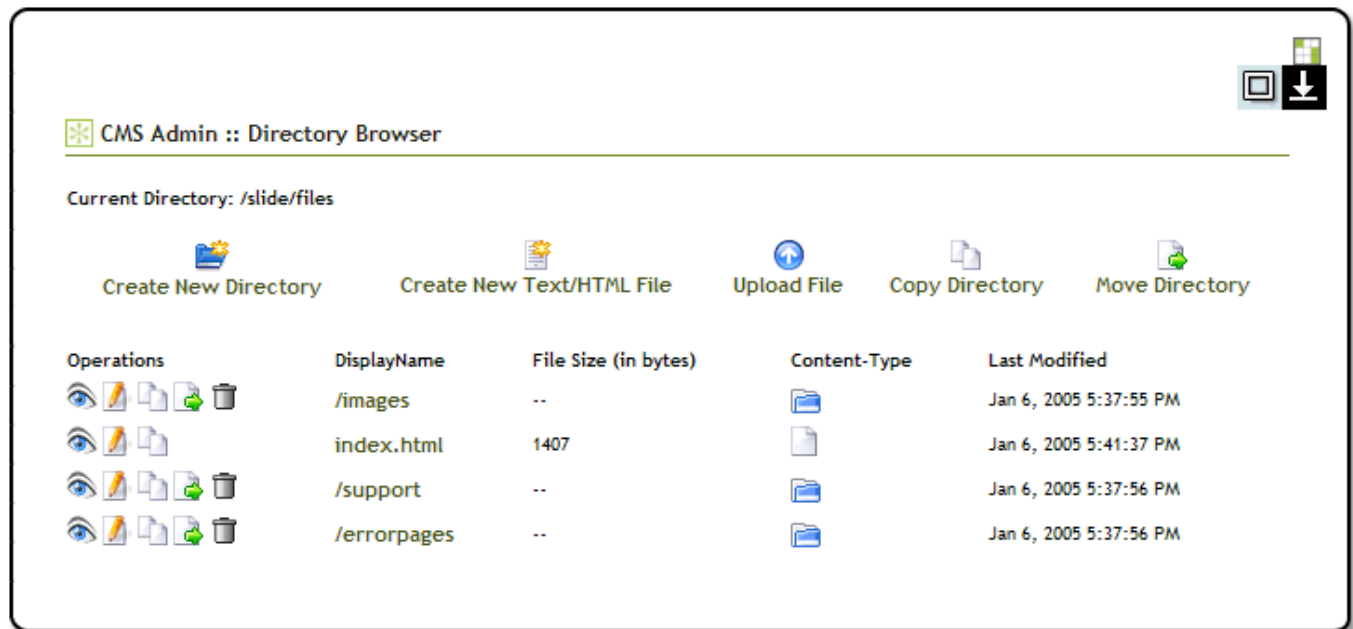
The AdminCMS Portlet allows control over the content management system.

Viewing the AdminCMS Portlet is accomplished by logging in as an admin (admin/admin) and navigating to the admin page <http://localhost:8080/portal/index.html?ctrl:id=page.default.admin>













You should then be presented by a page that is similar to this:








It is important for a user to note the action icons used throughout the portlet and their meanings. The action options change depending on what type of resource the user is dealing with. All possible actions are listed here:

-  - Launches HTML WYSIWYG Editor window for HTML files. Launches upload dialog window for binary type files.
-  - Opens the copy file/folder dialog window.
-  - Opens the move file/folder dialog window.
-  - Launches HTML WYSIWYG Editor window.
-  - Opens the create folder dialog window.
-  - Opens the upload file dialog window.
-  - Opens the delete confirmation dialog window.
-  - In the case of files, opens the file properties view. In the case of folders, opens the folder listing.
-  - Moves up the folder tree when clicked on.
-  - Expands directory tree.

Additionally, there are icons that help describe the types of resources present on the page:


-  - Denotes this resource as a binary file type.

-  - Denotes this resource as text/html or text/plain file types.
-  - Denotes this resource as a folder or "collection".

## 6.2. Functionalities

This section describes common actions a user can perform from within the AdminCMS Portlet.


### 6.2.1. Viewing Directories

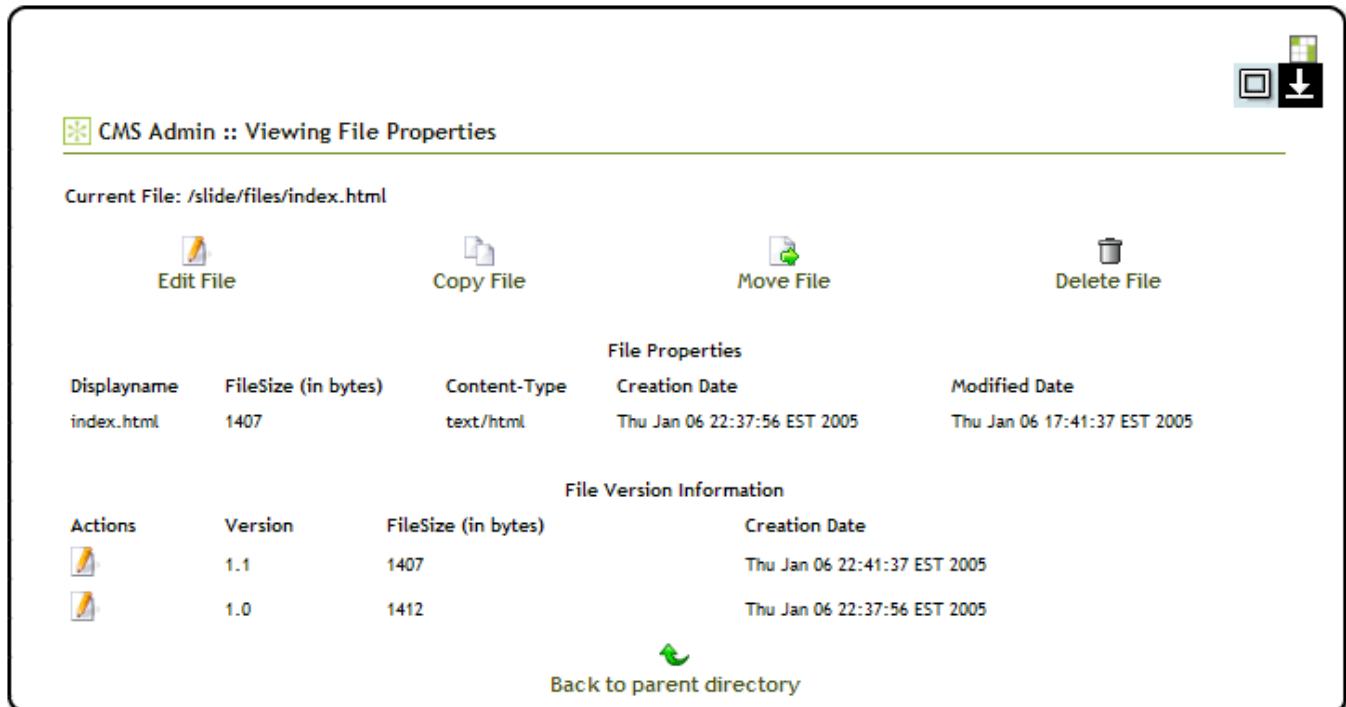
A user can list directory contents by either clicking on the  icon, or clicking on the directory's "DisplayName". All actions are possible from this screen.

#### Note

A user is not able to delete the homepage when in the content store's root directory. Nor is the user allowed to move above this directory.





### 6.2.2. Viewing File Properties

Clicking on the  icon or the "DisplayName" of a file brings up the File Properties page.






**CMS Admin :: Viewing File Properties**

Current File: /slide/files/index.html

 Edit File
  Copy File
  Move File
  Delete File

File Properties				
Displayname	FileSize (in bytes)	Content-Type	Creation Date	Modified Date
index.html	1407	text/html	Thu Jan 06 22:37:56 EST 2005	Thu Jan 06 17:41:37 EST 2005

File Version Information			
Actions	Version	FileSize (in bytes)	Creation Date
	1.1	1407	Thu Jan 06 22:41:37 EST 2005
	1.0	1412	Thu Jan 06 22:37:56 EST 2005


 Back to parent directory

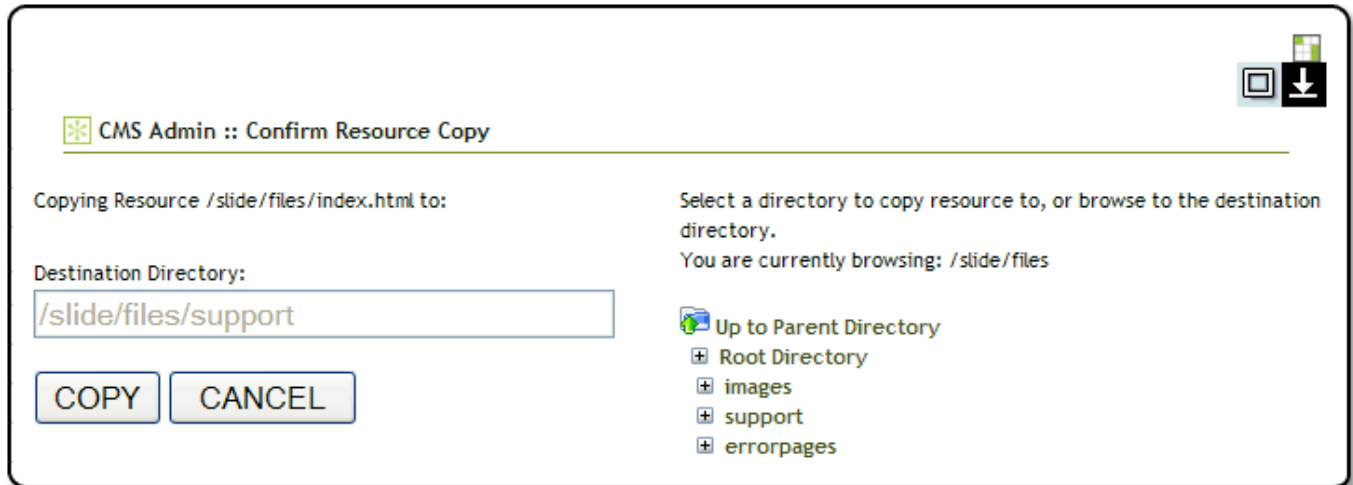
The File Properties window displays all the possible actions available to perform on a file. It displays certain properties of the file: DisplayName, FileSize, Content-Type, Creation Date, and when it was last modified.

Version Information is also contained on this screen. It is important to note that the top-most version in the list is

the current version that is "live". Editing an older version of a file and saving it, will auto-create a new version of the file and make it the current "live" version.

### 6.2.3. Copying Files/Directories

Clicking on the  icon displays the copy file/directory dialog window.



**CMS Admin :: Confirm Resource Copy**


Copying Resource /slide/files/index.html to:

Destination Directory:


Select a directory to copy resource to, or browse to the destination directory.

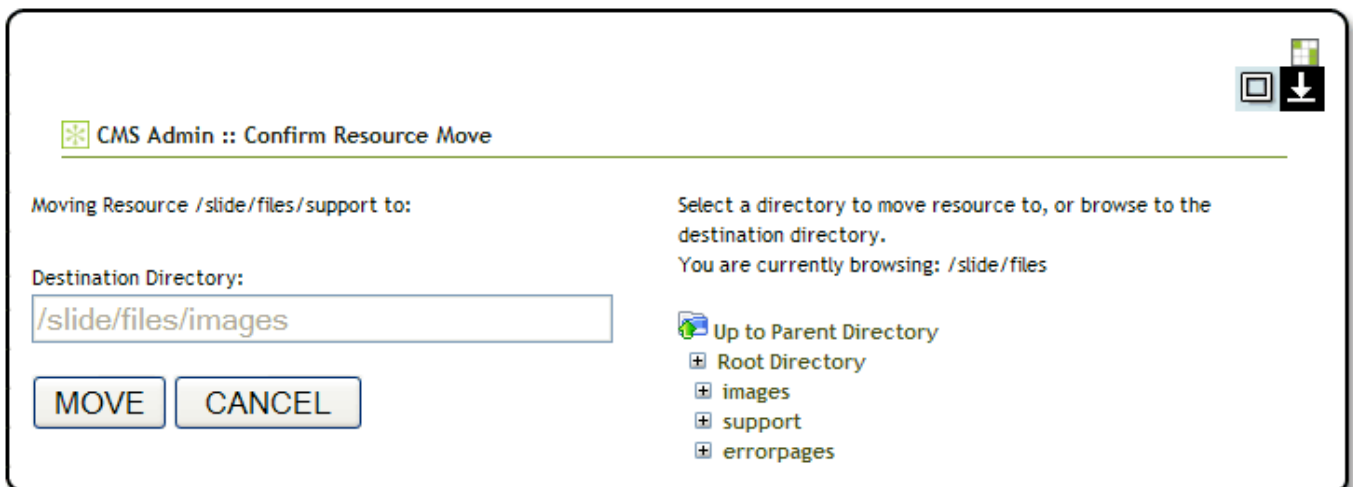
You are currently browsing: /slide/files

- Up to Parent Directory
- + Root Directory
- + images
- + support
- + errorpages

The copy resource window allows a user to copy files to any folder on the system, as well as copy whole directory trees to any directory on the system. A user can select which destination directory to copy the resource to, by using the directory browser. Clicking the  icon expands the directory tree. Clicking on the name of the directory within the tree, sets it as the destination directory for the copied resource.

### 6.2.4. Moving Files/Directories

Clicking on the  icon displays the move file/directory dialog window.



**CMS Admin :: Confirm Resource Move**


Moving Resource /slide/files/support to:

Destination Directory:


Select a directory to move resource to, or browse to the destination directory.

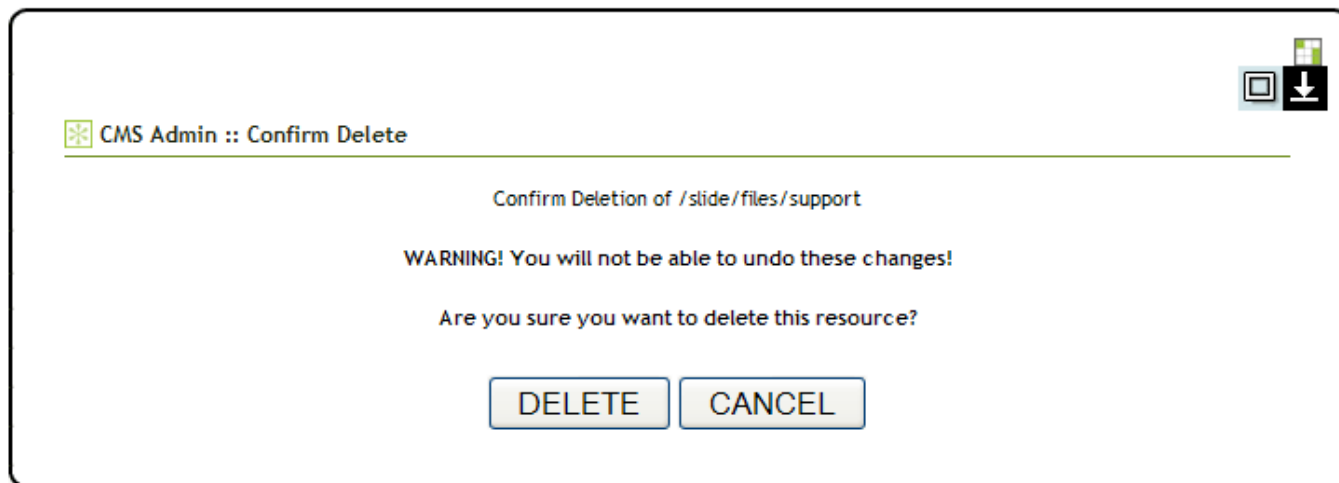
You are currently browsing: /slide/files

- Up to Parent Directory
- + Root Directory
- + images
- + support
- + errorpages

The move resource window allows a user to move files to any folder on the system, as well as move whole directory trees to any directory on the system. A user can select which destination directory to move the resource to, by using the directory browser. Clicking the  icon expands the directory tree. Clicking on the name of the directory within the tree, sets it as the destination directory for the moved resource.

## 6.2.5. Deleting Files/Directories

Clicking on the  icon displays the delete file/directory confirmation window.




The screenshot shows a dialog window titled "CMS Admin :: Confirm Delete". The window has a title bar with a close button and a download icon. The main content area contains the text "Confirm Deletion of /slide/files/support", a warning message "WARNING! You will not be able to undo these changes!", and a question "Are you sure you want to delete this resource?". At the bottom, there are two buttons: "DELETE" and "CANCEL".

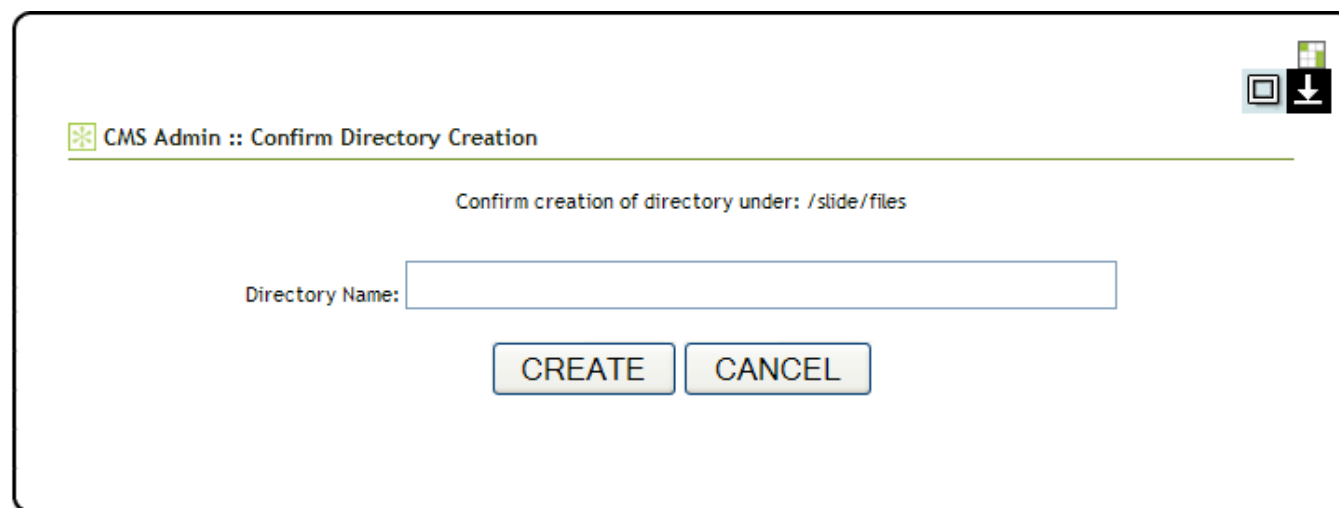
The delete resource confirmation window allows a user to delete a file, or a directory on the system. Note that deleting a directory, will delete the entire tree, so all directories under the deleted one, will also be deleted.

### Warning

Currently, there is no way to retrieve deleted files/directories! Deleting a file or directory is permanent!

## 6.2.6. Creating Directories


Clicking on the  icon displays the create directory dialog window.

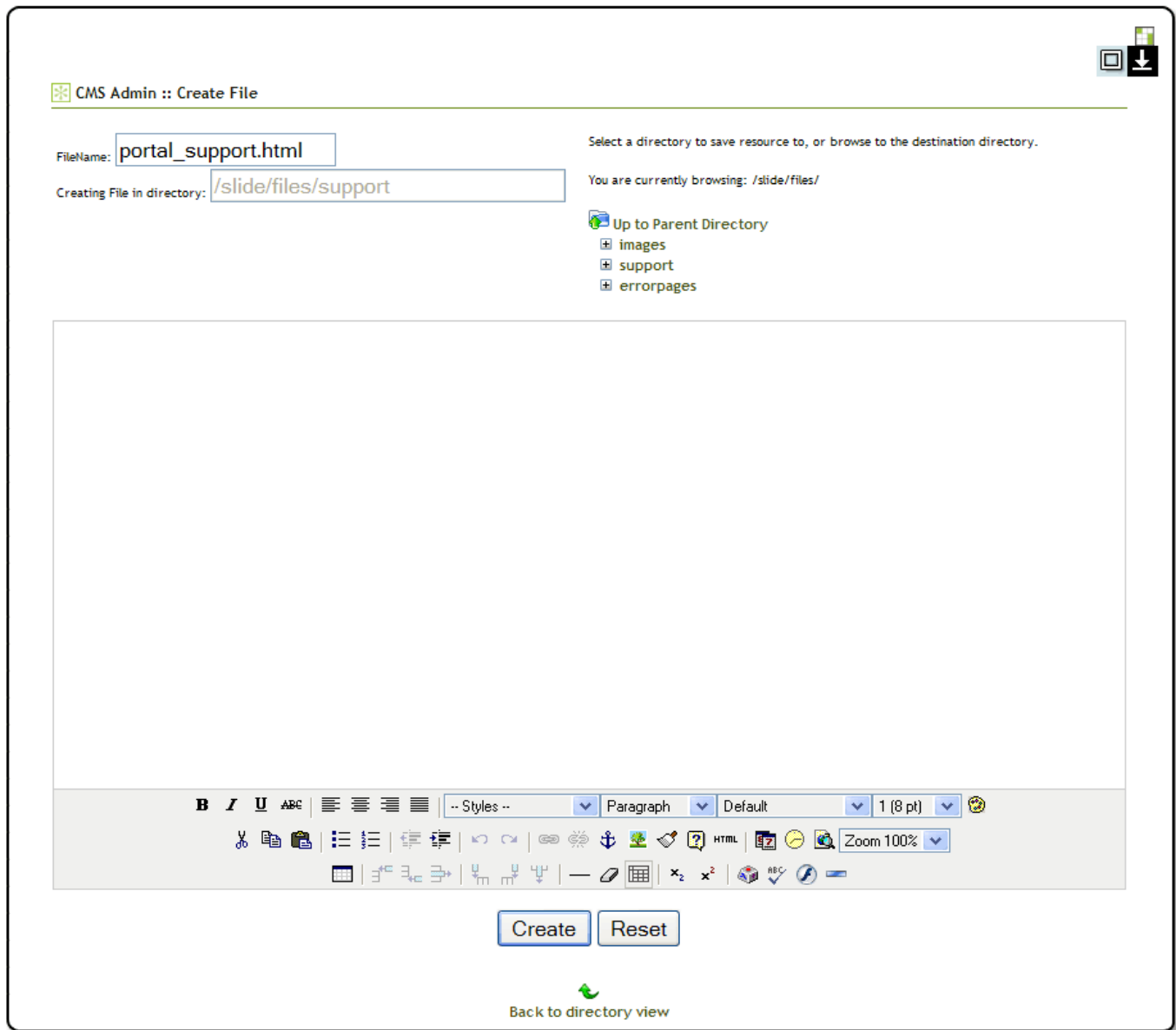




The screenshot shows a dialog window titled "CMS Admin :: Confirm Directory Creation". The window has a title bar with a close button and a download icon. The main content area contains the text "Confirm creation of directory under: /slide/files", a text input field labeled "Directory Name:", and two buttons: "CREATE" and "CANCEL".


The create directory resource window allows a user to create a directory under the one he originally clicked the icon in. On this window, a user can specify a name for the new empty directory.

## 6.2.7. Creating Text/HTML Files

Clicking on the  icon displays the create file dialog window with the embedded WYSIWYG editor and directory browser.




The create file window allows a user to create a text or HTML file using the embedded WYSIWYG HTML editor. The editor is a fully-functional HTML editor with a myriad of HTML functions. It also includes a preview  button and a source view  button.

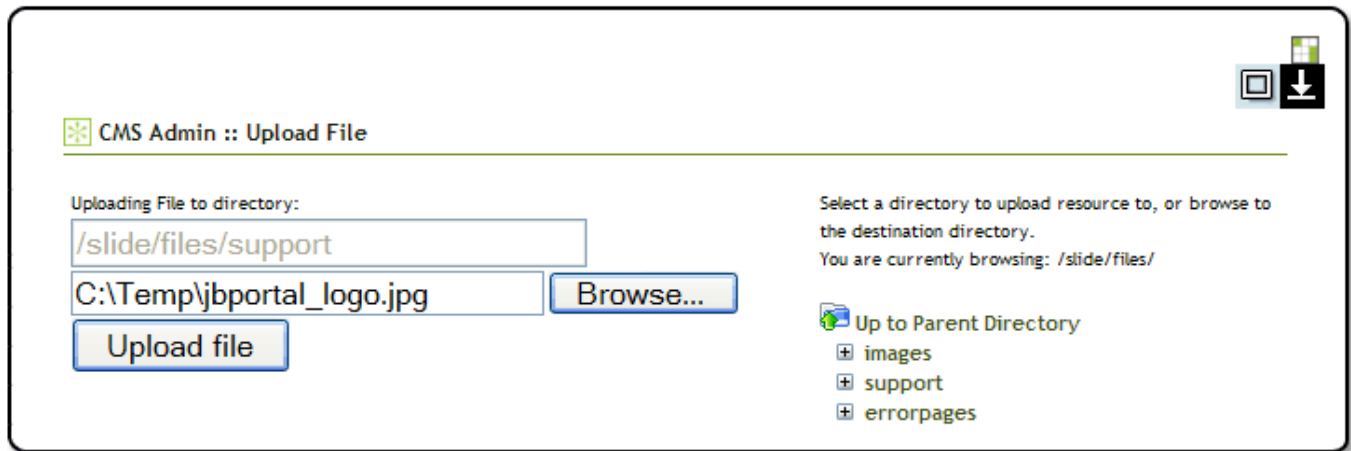
An in-depth walk-through of the editor is beyond the scope of this document. However, the editor does contain help pages within it, that can be accessed by clicking the  icon.


## Note

It is important to note here that when creating links to images or other resources within the system, as user must use the relative file path to that resource. ie: images/hello.gif. Keep in mind at all times that the document base is `http://localhost/portal/` by default!


## 6.2.8. Uploading Binary Files

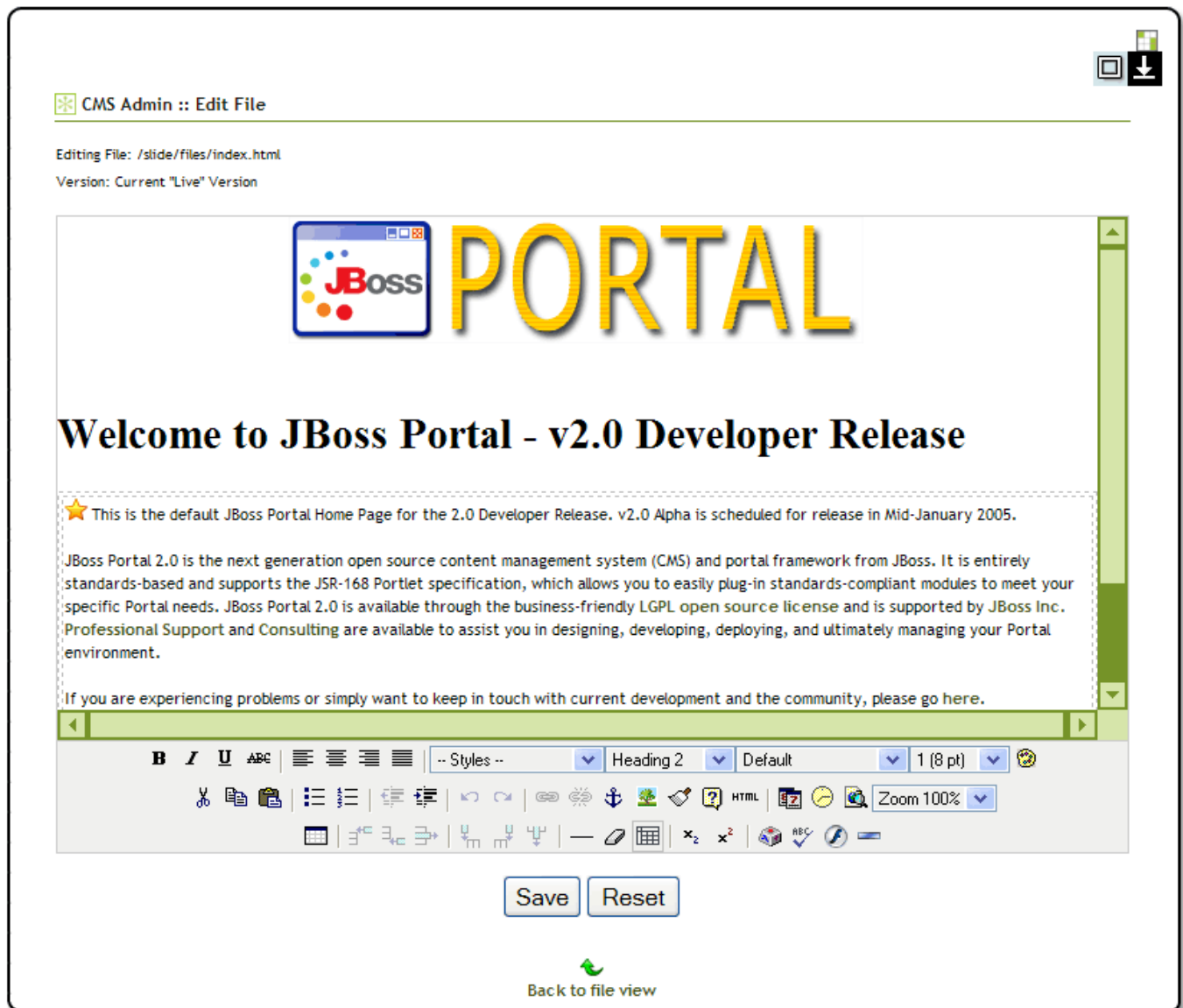
Clicking on the  icon displays the upload file dialog window.




The upload file window allows a user to upload files to any directory on the system. The upload process will work on files up to 1GB and of all types. A user can select which destination directory to upload the resource to, by using the directory browser. Clicking the  icon expands the directory tree. Clicking on the name of the directory within the tree, sets it as the destination directory for the uploaded resource.

## 6.2.9. Editing Text/HTML Files

Clicking on the  icon displays the edit file dialog window with the embedded WYSIWYG editor and directory browser.



The edit file window allows a user to edit a text or HTML file using the embedded WYSIWYG HTML editor. The editor is a fully-functional HTML editor with a myriad of HTML functions. It also includes a preview  button

and a source view  button.

## Note

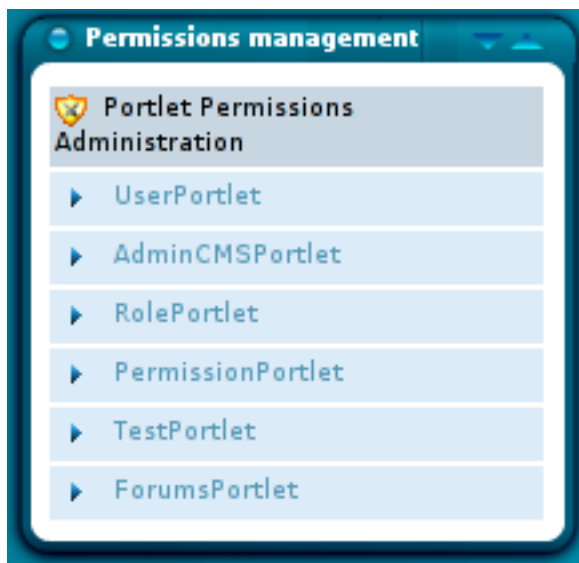
It is important to note here that when creating links to images or other resources within the system, as user must use the relative file path to that resource. ie: images/hello.gif. Keep in mind at all times that the document base is <http://localhost/portal/> by default!

## Permissions Portlet

Thomas Heute <theute@jboss.org>

### 7.1. Introduction

The permissions portlet is dedicated to manage permissions for portlets.



### 7.2. Functionalities

#### 7.2.1. Add a role to a permission

The portlet aims at attributing roles to a permission. Below is an example to setup permissions for the forums portlet.



**Permissions management**

**Modify Domain:**

Domain:

**Change Domain**

**Select a Component to Modify:**

- ▶ **Global Admin Privilege**

**Defined roles:** Administrators  
▶ [Add/Remove roles](#)  
**Implied roles:** Administrators,
- ▶ **Global deleting privilege**

▶ [Add/Remove roles](#)  
**Implied roles:** Administrators,
- ▶ **Global editing privilege**

▶ [Add/Remove roles](#)  
**Implied roles:** Administrators,
- ▶ **Global add privilege**

**Defined roles:** Users  
▶ [Add/Remove roles](#)  
**Implied roles:** Administrators, Users,
- ▶ **Global reading privilege**

**Defined roles:** Anonymous  
▶ [Add/Remove roles](#)  
**Implied roles:** Anonymous, Administrators, Users,

**Select a sub-component to define its permissions:**

- ▶ [Dummy demo category \[Categories management\]](#)

 **Back**

At the very top, a domain can be specified. A domain is defined by the portlet, it can be the name of a portal for example. It gives the scope of the permissions. The first screen gives you the top level privileges, usually it will show global permissions.

Defined roles gives the list of roles that has been defined for the corresponding permission while implied roles will show all the roles that are implied. For example in the forums, the admin role implies the add permission, defining the role "Admins" on the Admin permission, would show "Admins" in the implied roles of the add permission. Only defined roles can be removed so if a permission need to be restricted, the most permissive permission has to be removed before a less permissive one is added.

Under the global permissions, there is a list of sub-components for a finer grain permission definition. For example,

the forums portlet gives the ability to define permissions on a single category. By clicking on the category, the following screen would show up:

The screenshot shows a web interface titled "Permissions management". It has a light blue header bar with a shield icon and the title. Below the header, there's a section "Modify Domain: Dummy demo category" with a "Domain:" label and a text input field, followed by a "Change Domain" button. The main area is titled "Select a Component to Modify:" and contains a list of components, each with a right-pointing arrow icon and a link to "Add/Remove roles". The components are: "Admin a category" (Implied roles: Administrators), "Delete a forum in the category" (Implied roles: Administrators), "Add a forum in the category" (Implied roles: Administrators), "Edit a forum in the category" (Implied roles: Administrators), and "Read a category" (Implied roles: Anonymous, Users, Administrators). Below this list is a section "Select a sub-component to define its permissions:" with two items: "Second forum [Forums security management]" and "First forum [Forums security management]". At the bottom center, there is a green circular arrow icon and the text "Back".

**Permissions management**

Modify Domain: Dummy demo category

Domain:

**Change Domain**

Select a Component to Modify:


- ▶ **Admin a category** ▶ Add/Remove roles  
Implied roles: Administrators,
- ▶ **Delete a forum in the category** ▶ Add/Remove roles  
Implied roles: Administrators,
- ▶ **Add a forum in the category** ▶ Add/Remove roles  
Implied roles: Administrators,
- ▶ **Edit a forum in the category** ▶ Add/Remove roles  
Implied roles: Administrators,
- ▶ **Read a category** ▶ Add/Remove roles  
Implied roles: Anonymous, Users, Administrators,


Select a sub-component to define its permissions:

- ▶ Second forum [Forums security management]
- ▶ First forum [Forums security management]

Back


The forums security model dictates that the global admin permission implies the category admin permission, and the global add permission implies the category read permission, with that in mind, the implied roles should make sense. If you select a sub-component, you can define an even finer-grain permission and the following screen will show up:


 **Permissions management**


 **Modify Domain:** Dummy demo category:First forum


Domain:


**Change Domain**


 **Select a Component to Modify:**


 **Admin a forum**


 [Add/Remove roles](#)  
**Implied roles:** Administrators,


 **Delete a post in a forum**


 [Add/Remove roles](#)  
**Implied roles:** Administrators,


 **Moderate a forum**


 [Add/Remove roles](#)  
**Implied roles:** Administrators,


 **Post in a forum**


 [Add/Remove roles](#)  
**Implied roles:** Administrators, Users,

 **Edit a post in a forum**

 [Add/Remove roles](#)  
**Implied roles:** Administrators,

 **Read a forum**

 [Add/Remove roles](#)  
**Implied roles:** Anonymous, Administrators, Users,

  
Back

## Forums Portlet

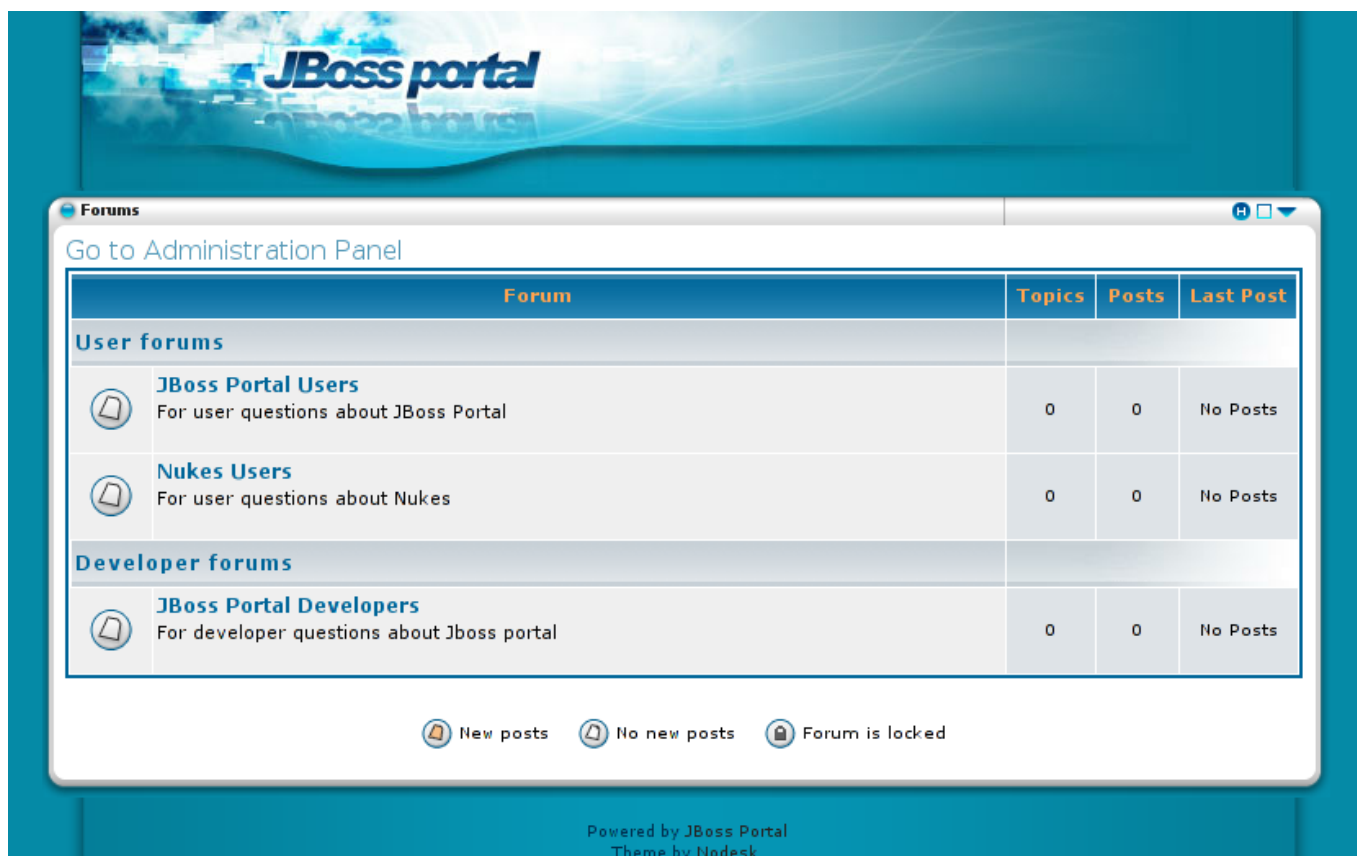
Thomas Heute <theute@jboss.org>

### 8.1. Introduction

#### Warning

The forums portlet is GPL licensed

The forums portlet is a port of the phpBB forums as a Java portlet. It is packaged independently of the core, so you can easily use it or not depending on your own needs.



Above is the main window displayed by default to any user. It lists all the forums classified by categories. It is possible to see how many topics and posts were written for each forum and the date and user of the last post. All those categories and forums can be configured if the user has the correct privileges. The next image shows the main administration interface available to users with the correct credentials.

**Forum Administration**

From this panel you can add, delete, edit, re-order and re-synchronise categories and forums

Forum Administration						
<b>User forums</b>				Edit	Delete	Move up Move down
JBoss Portal Users	0	0	Edit	Delete	Move up	Move down
For user questions about JBoss Portal						
Nukes Users	0	0	Edit	Delete	Move up	Move down
For user questions about Nukes						
<input type="text"/> <input type="button" value="Create new forum"/>						
<b>Developer forums</b>				Edit	Delete	Move up Move down
JBoss Portal Developers	0	0	Edit	Delete	Move up	Move down
For developer questions about Jboss portal						
<input type="text"/> <input type="button" value="Create new forum"/>						
<input type="text"/> <input type="button" value="Create new category"/>						

Powered by JBoss Portal  
Theme by Nodesk

## 8.2. Functionalities

User features:

- See the list of forums
- Post a new topic
- Read a topic
- Reply to an existing post
- Fast-reply to an existing post in the same page as the thread
- Email notification
- Quote a existing forum posts

Admin features:

- Create a new category of forum

- Edit the name of a category
- Delete a category and move the content to another category
- Create a new forum
- Edit the name and description of a forum
- Delete a forum and move the content to another forum
- Classify categories
- Classify forums
- Edit forum posts

## 8.3. Installation

If you are deploying from **binary**, just move portal-forums.ear in to your deploy directory.

If you are deploying from **source**:

To install forums, you need to go to the directory `forums` and type `sh build.sh deploy` it will create a file `portal-forums.ear` and copy it to `$JBoss_HOME/server/default/deploy`. If JBoss is already running you have nothing to do but to go to a page where the forums should be displayed (see your configuration).

To have the mail notification working, make sure that you correctly configure the mail service with an existing SMTP account in the file:  
`$PORTAL_HOME/core/src/resources/portal-server-war/WEB-INF/jboss-service.xml`

## 8.4. Configuration

In `$FORUMS_HOME/src/resources/portal-forums-war/WEB-INF/portlet.xml` you can configure the following options:

- `floodInterval` : Minimal time in seconds between two messages by a user.
- `fromAddress` : Email address appearing in the `From` field of notification emails.

## 8.5. Security

You can restrict access to the forums for certain roles, to do so edit the file `$FORUMS_HOME/src/resources/portal-forums-war/WEB-INF/jboss-portlet.xml`. You should see the existing part:

```
<scheme>
<domain></domain>
```

```
<item>
<path>/</path>
<permission>
<permission-name>Add</permission-name>
<role-name>Users</role-name>
</permission>
<permission>
<permission-name>Admin</permission-name>
<role-name>Admins</role-name>
</permission>
<!-- For non logged users -->
<permission>
<permission-name>Read</permission-name>
<role-name></role-name>
</permission>
</item>
</scheme>
```

This means that a user with role Users has the permission to add posts in forums, a user with role Admins has the permissions to Admin anything, while an anonymous user (not logged on), can only read.

If you want users to only view a category named "myCategory" to a certain role "myRole", here is an `item` that you can add:

```
<item>
<path>/myCategory</path>
<permission>
<permission-name>ReadCategory</permission-name>
<role-name>myRole</role-name>
</permission>
</item>
```