



Xelsius Portlet User Guide



Copyright

© 2008 Business Objects. All rights reserved. Business Objects owns the following U.S. patents, which may cover products that are offered and licensed by Business Objects: 5,555,403; 5,857,205; 6,289,352; 6,247,008; 6,490,593; 6,578,027; 6,831,668; 6,768,986; 6,772,409; 6,882,998; 7,139,766; 7,299,419; 7,194,465; 7,222,130; 7,181,440 and 7,181,435. Business Objects and the Business Objects logo, BusinessObjects, Business Objects Crystal Vision, Business Process On Demand, BusinessQuery, Crystal Analysis, Crystal Applications, Crystal Decisions, Crystal Enterprise, Crystal Insider, Crystal Reports, Desktop Intelligence, Inxight, the Inxight Logo, LinguistX, Star Tree, Table Lens, ThingFinder, Timewall, Let there be light, Metify, NSite, Rapid Marts, RapidMarts, the Spectrum Design, Web Intelligence, Workmail and Xcelsius are trademarks or registered trademarks in the United States and/or other countries of Business Objects and/or affiliated companies. All other names mentioned herein may be trademarks of their respective owners.

Third-party Contributors

Business Objects products in this release may contain redistributions of software licensed from third-party contributors. Some of these individual components may also be available under alternative licenses. A partial listing of third-party contributors that have requested or permitted acknowledgments, as well as required notices, can be found at: <http://www.businessobjects.com/thirdparty>

2008-02-27

Overview

An Xcelsius Portlet enables the integration of an Xcelsius visualization exported to Shockwave Flash Object (SWF) files into a portal server, making default values configurable and enabling one Xcelsius Portlet to communicate with another.

This document describes how to create and configure an Xcelsius Portlet from any Xcelsius visualization file.

Introduction

For each Xcelsius visualization to be published on a portal, a new Xcelsius Portlet is created. The Xcelsius Portlet discussed in this document is a template used to build each of these specific Xcelsius portlets.

While the Xcelsius Portlet complies with the JSR 168 standard, and can be deployed on any portal server, inter-portlet communication is currently only supported on IBM WebSphere Portals.

The Xcelsius Portlet was tested on IBM WebSphere Portal 5.1. All portlet management instructions in this guide are based on WebSphere Portal 5.1. All portal screenshots are taken from a WebSphere Portal 5.1 installation.

Xcelsius Portlet Generation Utility

The Xcelsius Portlet Generation Utility creates an Xcelsius Portlet for a given Xcelsius visualization file.

The Xcelsius Portlet Generation Utility's output is an Xcelsius portlet contained in a web application archive (.war), which can be installed on a portal server.

Installation

The Xcelsius Portlet Generation Utility does not require an installation. It only needs to be extracted to a convenient location. The only software needed is a Java Runtime Environment version 1.4.2 or higher.

Using the GUI

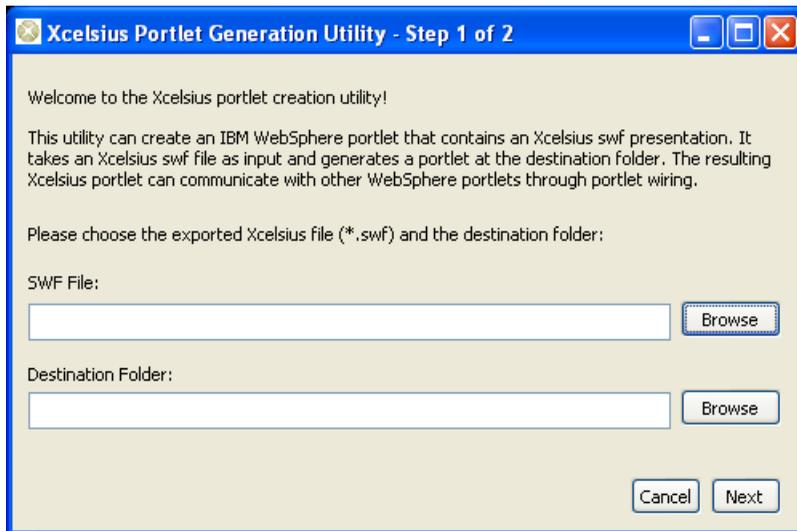
Starting the application

The Xcelsius Portlet Generation Utility is started by double-clicking Xcelsius PortletGenerationUtility.bat (.sh on Linux platforms).

Selecting a file and a destination

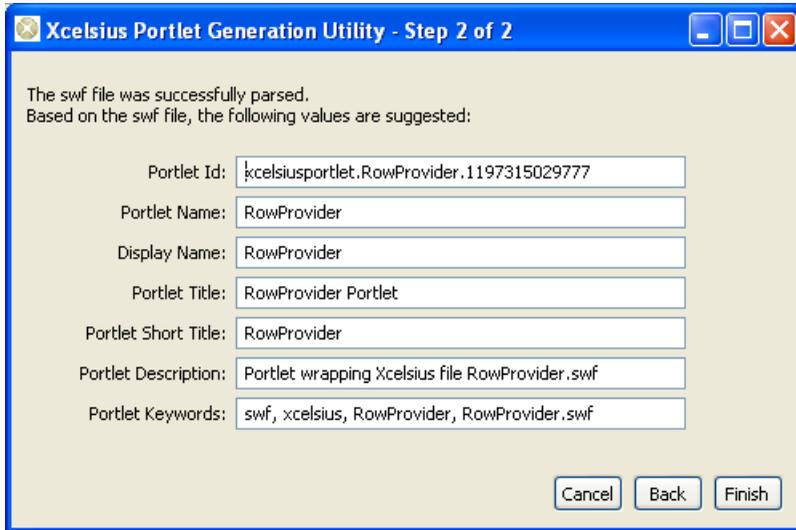
The first screen dialog is used to select the Xcelsius visualization for which the Xcelsius Portlet is created, and to decide where the resulting web application archive (.war) file should be stored.

Be sure that the file you select contains "Portal" connection(s). For details about the Xcelsius data connections that have special functionality in the Xcelsius portlets, see [Relevant Xcelsius Data Connections](#) on page 9 .



Inserting portlet specific data

Based on the visualization SWF file, suggested values are generated for each field. The second dialog allows modifying these values. The following table explains what these values are in the portlet context.



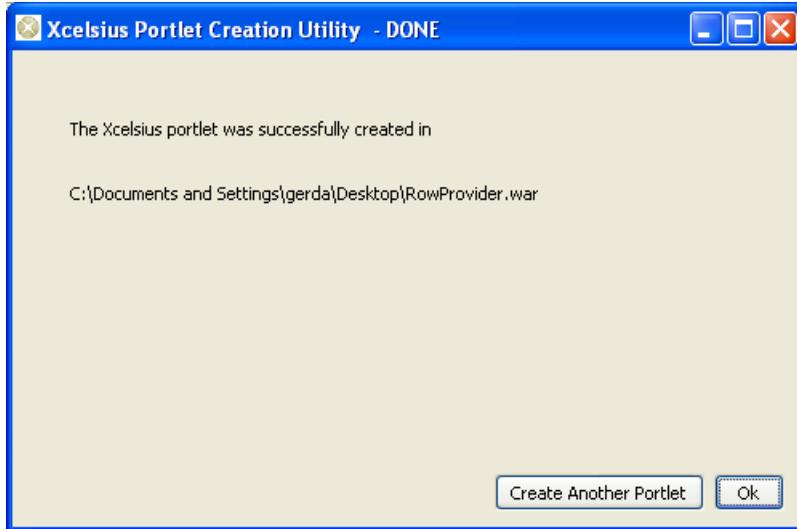
Field	Description
Portlet Id	Internal identification used by WebSphere Portal. The Portlet Id must be unique. Allowed characters are letters, digits, dot (".") and underscore ("_").
Portlet Name	The portlet's name, which also determines the name of the web application the utility generates. Allowed characters are letters, digits, dot (".") and underscore ("_").
Display Name	Appears when portlet is displayed.
Portlet Title	The portlet title is usually displayed in the title bar of the portlet's skin. Locale dependent.
Portlet Short Title	May be used instead of the portlet title. Locale dependent.
Portlet Description	The portlet description is displayed in the portal administration. (Keep it short; there's not too much room in there.)
Portlet Keywords	The portlet's keywords. Locale dependent.

Until the Finish button is pressed, nothing is written to the file system. Only then, the Portlet Generation Process is invoked.

Final screen

After the portlet has been created, the final dialog gives the absolute location of the generated web application.

To create another Portlet, click Create Another Portlet; otherwise click Ok to close the application.



Using the Command Line

The command line tool is an option for experienced users who want to create Xcelsius Portlets in a single step. It also enables scripting to create multiple Xcelsius Portlets automatically.

Note: Unlike the graphical version, the command line tool is platform independent.

Creating a portlet

On a command line, start the tool by typing the following:

```
java -jar XPGU_CommandLineTool.jar <*.swf file>  
<options>
```

Be sure to specify the "<*.swf file>" in a way that it can be found by the tool - i.e. give either the full path or the path relative to the location from which

you start the command line tool. The following table describes possible options.

Option	Description
-d directory	The destination folder.
-desc "string"	The portlet's description, which is used in the portal administration.
-portletid string	The portlet id must be unique. Allowed characters for a portlet id are letters, digits, dot ("."), and underscore ("_").
-portletname string	<p>1. The portlet name also determines the name of the web application.</p> <p>Allowed characters for a portlet name are letters, digits, dot ("."), and underscore ("_").</p>
-displayname string	The display name.
-keywords string	Keywords
-portlettitle string	The title that shows up on the portal page.
-portletshorttitle string	The short title.
-ssoprincipal classname	Name of the principal class used for SSO.
-sso timeout string	Timeout for the SSO token.
-sso nooflogons number	Number of logons for which the token is valid.
-properties filename	<p>Non-default properties file (useful if you have multiple configurations-encapsulate all settings in one properties file instead typing them every time)</p> <p>Be sure that your properties file contains all the values the default properties file contains.</p> <p>Values specified as command line options always override values in the properties file.</p>

Option	Description
-template war-file	If a template web application other than the default should be used, you can specify it using this parameter.

Caution: For all options, be sure to use quotation marks if your values contain spaces (" ").

Note: Using the options "-properties" and "-template" will not be necessary under normal circumstances for normal users.

Xcelsius Portlet Generation Utility Configuration

Both the template web application and its default settings can be configured.

Note: The name of the utility's JAR file may vary, depending on the platform and graphical or command line tool.

Xcelsius Portlet Generation Utility Configuration

The properties file

The Xcelsius Generation Utility's property file name is XcelsiusPortletCreationUtility.properties. To modify it, you need to unpack the utility's JAR file, edit it, and preferably put it back into the JAR file. Alternatively, it can be kept in the same folder as the JAR file.

The properties file contains default values used for Xcelsius Portlet parameters. These default values are used to generate the suggested values displayed in dialog Step 2 of 2 in the graphical version of the utility, and if no special options are specified, in the command line tool version.

It also contains values that are not expected to change frequently, such as which template web application file to use and the single sign-on authentication strategy.

The template web application

The files in the template application contain placeholders for special values, which are replaced by the utility. Make sure you keep all of the files and placeholders. When the utility does not find files or placeholders in the files, it assumes that the template web application has been corrupted and will not create any Xelsius Portlets.

If for some reasons (e.g. different texts or layout on the Xelsius Portlet's JSPs, adding special supported languages etc) you need a different template web application, we recommend modifying the original WAR file from the utility (template.war).

Relevant Xelsius Data Connections

When creating an Xelsius visualization, there are two connection types that have special functionality if used in an Xelsius Portlet: Portal Data Connection and Web Service Connection. These connections are in the Data Manager. Portal Data connections can be of the following types: None (parameter), Consumer or Provider.

Each Xelsius visualization may optionally contain one provider connection or one consumer connection. It can also contain neither.

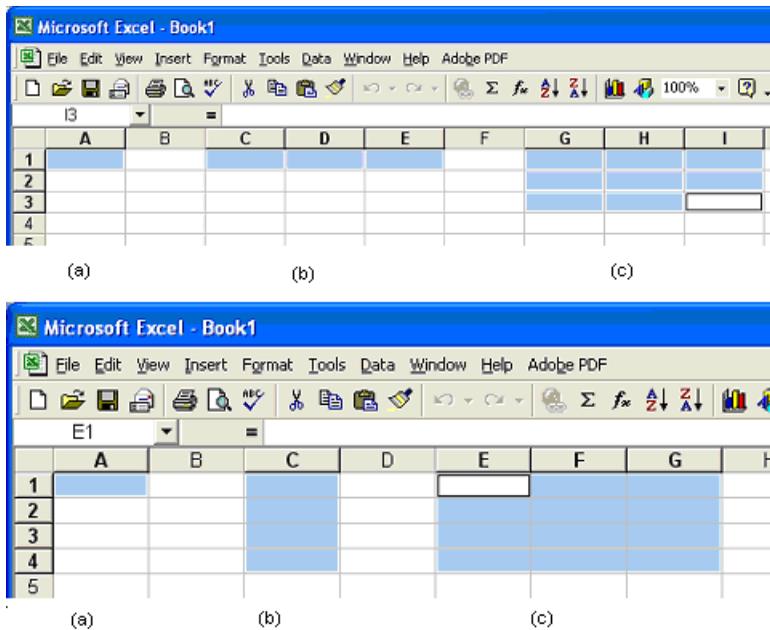
Parameter and Web Service Connections are independent from consumers and providers. Their numbers are not limited.

Portal Data Consumer

There are three different kinds of consumer, distinguished by the range they cover:

- Cell: consumes data and maps it to a single cell.
- Row: consumes data and maps it to a one dimensional group of cells.
- Table: consumes data and maps it to a two dimensional group of cells.

The image below illustrates how ranges for a Cell Consumer, a Row Consumer, and a Table Consumer are selected. Note that for the Row Consumer, the selected range may be vertical (row) or horizontal (column).



Portal Data Provider

As with the consumers, there are three different types of providers:

- Cell: transmits values of a single cell
- Row: transmits values of a one dimensional group of cells
- Table: transmits values of a two dimensional group of cells

Portal Data Parameters (No Connection)

Portal Data Parameters enable you to modify values from outside the Xcelsius SWF visualization.

Their values can be modified in the Xcelsius Portlet's edit mode.

To be able to use them in the Xcelsius visualization, the parameters must be linked to a cell in the spreadsheet.

When linked to a cell, the parameter is given a default value. If the cell is empty, the default value will be zero (0) .

There are different types of parameters. The type indicates which values are expected. It does not limit you in what you might be doing with these parameters.

- Text parameters allow you to enter any text.
- Number parameters allow you to enter numbers.
- Check Box parameters allow you to enter exactly two values: checked, represented by (1), and unchecked, represented by (0).
- Date parameters represent a date.
- List Box parameters allow you to specify a number of options among which the user may later choose.

For details on how these values can be modified in the portal, see Section "Xcelsius Portlet User Interface".

User Information

Text parameters with special names will be translated to information about the portal user using the portlet. This functionality can be used to personalize the Xcelsius visualization, e.g. by using the user's name in a welcome message.

Portal Data Parameters whose name starts with "USER_INFO." will not show up in the Xcelsius Portlet's edit mode. Be sure not to let your Parameters' names start with "USER_INFO." if they are not intended to pick up user data.

The following table shows supported values for "USER_INFO." names. Note that not all fields will contain values on every portal. Check with your portal administrator which values are supported by the portal's user registry.

Special Values
USER_INFO.user.name.given
USER_INFO.user.name.family
USER_INFO.user.name.prefix
USER_INFO.user.name.middle

Special Values
USER_INFO.user.name.employer
USER_INFO.user.name.department

Web Service Connection

The Xcelsius visualization can get data from a web service. If this web service is created using "Query As A Web Service", it will connect to the BusinessObjects Enterprise server specified in the URL of the Web Service Connection. See the section below on configuring Single Sign-On.

Xcelsius Portlet

Once an Xcelsius Portlet is created, it is bound to its underlying visualization, and its portlet properties like name, title etc are set. It is ready to be installed on a portal server.

The only thing left to do is modifying the Xcelsius Portlet's Portal Data Parameters, set up its collaborations with other portlets, and some administrative tasks.

This section covers the administrative tasks needed to get a portlet up and running, the Xcelsius Portlet from a user's perspective, and how collaboration in a running portlet works - everything you'd need to know if the Xcelsius Portlet should collaborate with other, non-Xcelsius portlets.

Xcelsius Portlet Administration

Before an Xcelsius Portlet can be used on a portal page, a user with administrator privileges has to install it.

Configuration is only needed for inter-portlet communication and, in most scenarios, access rights settings. Settings for Single Sign-On are covered in the next section.

Installing an Xcelsius Portlet

Each Xcelsius portlet is packaged in its own web application. It does not require any special settings or additional libraries.

Configuring Inter-Portlet Communication

Xcelsius Portlets may transmit data to other portlets. This is set up in the portal's administration, under the Manage Pages section. Every portal user who has the right to edit the page can configure inter-portlet communication.

Choose the page on which the portlets that should communicate are published and select edit.



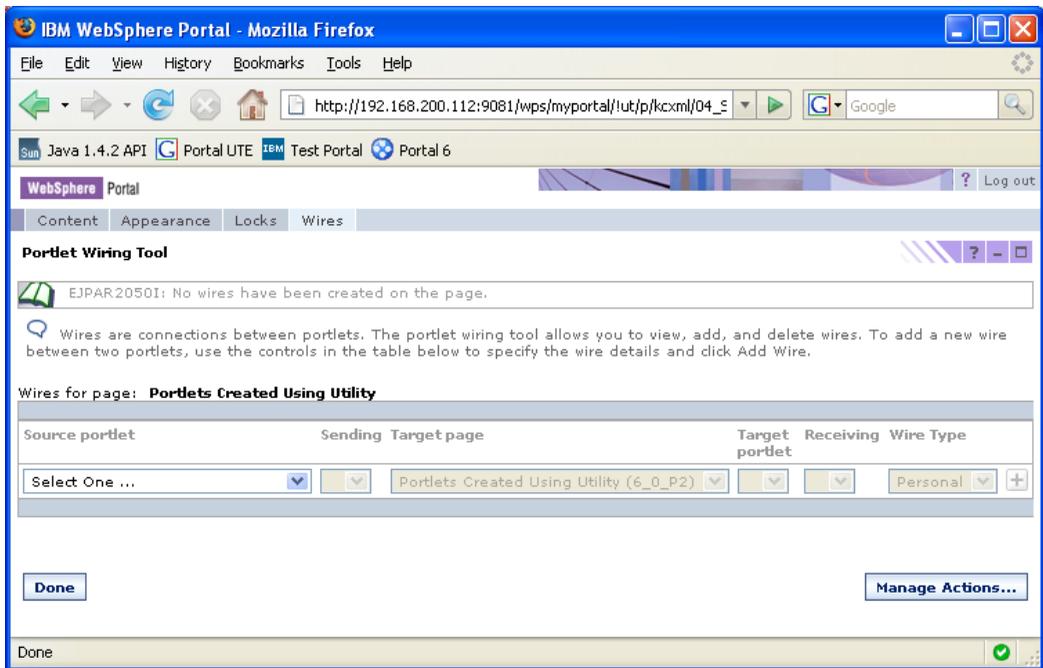
In the edit page section, select the rightmost section titled "Wires". In the Source portlet drop down, select the portlet that holds the provider. Select the field that should be sent. For Xcelsius Portlets, there will be only one option in the drop down box.

Select the page on which your consumer portlet is published as the Target page. Select your consumer portlet as Target portlet. Again, for Xcelsius portlets, there is only one option in the Receiving drop down box.

Choose the wire type and click the "+" button to finalize the wire creation. Your portlets are now able to communicate.

Caution: Be aware that portlets that contain only consumers also show up as potential source portlets, and portlets containing only a provider also show up as possible target portlets. This is because a portlet either is wiring-enabled or not. For WebSphere Portal, there is no distinction between provider and consumer portlets. If you choose a consumer portlet as the provider, this wire will never be used, because the consumer will never send any data.

Caution: Be sure that the mapping of providers and consumers is compatible with your intent. Generally, let Cell Providers communicate with Cell Consumers, Row Providers with Row Consumers, and Table Providers with Table Consumers. Otherwise, you might not see the intended results.



Xcelsius Portlet User Interface

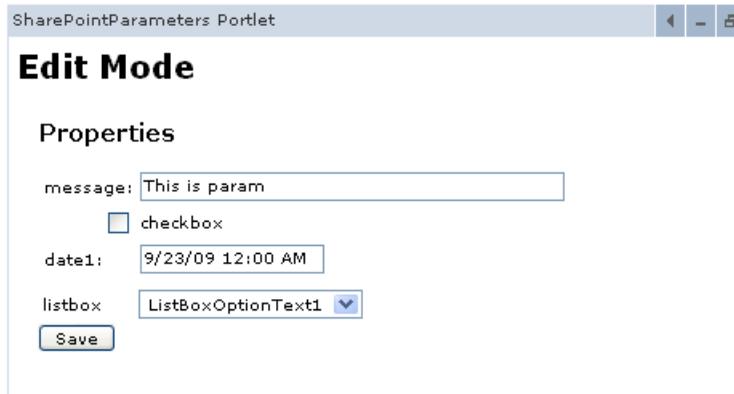
The Xcelsius Portlet will have the default height and width of the Xcelsius visualization it integrates. It displays the SWF file using the web browser's Adobe Flash Player plug-in.

Modifying Xcelsius Data Portal Parameters

Different types of the visualization's Portal Data Parameters render to different input fields in the Xcelsius Portlet's edit mode. The following table contains the html control for each parameter type.

Parameter Type	Edit Mode Control
Text	Input Field
Number	Input Field
Date	Input Field
Check Box	Check Box
List Box	Drop-Down Box

This image shows the edit mode of an Xcelsius Portlet with a text parameter named "message," a check box parameter named "checkbox," a date parameter named "date1," and a list box parameter named "listbox".



Xcelsius Data in the Xcelsius Portlet

Data from the Xcelsius visualization file comes in contact with the Xcelsius Portlet in two ways:

- Data transmitted between Xcelsius Portlets connected through wires.
- Data for parameters set by the user in the portlet's edit mode.

Portal Data Providers and Portal Data Consumers exchange data in XML format. The Xcelsius Portlet does not modify the data that is transmitted. This table shows the different values.

<pre><data> <row> <column> DATA </column> </row> </data></pre>	<pre><data> <row> <column> DATA </column> <column> DATA </column> <column> DATA </column> </row> </data></pre>	<pre><data> <row> <column> DATA </column> <column> DATA </column> </row> <row> <column> DATA </column> </row> </data></pre>
--	--	---

Data that goes into the Xcelsius visualization – data received by a consumer and data set for a parameter – is set as a flash variable.

For consumers, the Xcelsius Portlet does not impose any restrictions on the sender of the data, meaning the sending entity does not have to be another Xcelsius Portlet. If you need to transmit data from another, non-Xcelsius portlet, be sure to wrap the data as described in the table and set up the wires similar to how wires are set up between Xcelsius Portlets. It is equally possible to send data from an Xcelsius Portlet to a non-Xcelsius portlet.

Note: If transmitting data between a non-Xcelsius Portlet provider and an Xcelsius Portlet consumer works, but the received data is not correctly formatted in the Xcelsius Portlet, check if your data is enclosed in XML as described by the table.

Single Sign-On

Xcelsius visualizations may consume data sources at a BusinessObjects Enterprise environment, through a Query As A Web Service connection.

The Xcelsius Portlet supports Single Sign-On to a BusinessObjects Enterprise environment by creating tokens that are passed to the Flash player when the exported Xcelsius file is loaded.

These tokens authenticate the user. If authentication fails, the user has to enter username and password in the Flash animation.

Prerequisites

For Single Sign-On to work, a special login module must be installed and configured correctly.

If that is not already done, the following steps are necessary to install the login module.

Optional: Configure the WebSphere Portal Server for Trusted Authentication

1. Click Environment > Manage WebSphere Variables
2. Click New
3. Set the Name to BOBJ_HOME, and Value to the head of your BOE environment. If there is no BOE environment installed on the machine, choose any directory you like, e.g. C:\BusinessObjects.
4. Create a text file named TrustedPrincipal.conf under `$(BOBJ_HOME)/<platform>/plugins/auth/secEnterprise`.

Insert the same shared secret you set on the CMS here, e.g.
SharedSecret=Xcelsius

Add the JAR files to the class library

1. Copy the necessary libraries to the portal server, e.g. to `$(WPS_HOME)/Portal_home/shared/app/BOLoginModule`.
2. Start the Application Server, and open the Application Server Admin Console.
3. Click Environment > Shared Libraries.
4. Click New.

5. Set the Name to BOLoginModule, and the Classpath to

```
{WPS_HOME}/shared/app/BOLoginModule/XELoginModule.jar  
{WPS_HOME}/shared/app/BOLoginModule/cecore.jar  
{WPS_HOME}/shared/app/BOLoginModule/ceplib.jar  
{WPS_HOME}/shared/app/BOLoginModule/ceplugins.jar  
{WPS_HOME}/shared/app/BOLoginModule/cesession.jar  
{WPS_HOME}/shared/app/BOLoginModule/ebus405.jar  
{WPS_HOME}/shared/app/BOLoginModule/corbaidl.jar
```

6. Click Ok.

7. Click Servers > Application Servers > WebSphere_Portal.

8. In the Additional Properties, select Classloader.

9. Click on the Classloader's Id. In the Additional Properties, select Libraries. Click Add.

10. Select BOLoginModule and click Ok.

Configuring the Login Module

1. Click Security > JAAS Configuration > Application Logins.

2. Click on Portal_LTPA.

3. In Additional Properties, click on JAAS Login Modules.

4. Click New.

5. Set the Module Classname to com.businessobjects.security.jaas.XELoginModule. If it is not inserted by the server automatically, set the proxy class name to com.ibm.ws.security.common.auth.module.proxy.WSLoginModuleProxy.

Set the Authentication Strategy to OPTIONAL.

Click Apply.

6. Click on Custom Properties for the newly added Login Module. The following properties can be set:

- DefaultCMS
- DefaultAuthentication: set to TrustedAuthentication for trusted authentication, or secEnterprise for standard password authentication. This settings must match the strategysset in the CMS

- SessionTimeout: 120
- LogLevel: Value from 0 (logging turned off) to 7 (highest)
- LogFile: Path to the log file, e.g. C:\auth.log

Note: The property values are case sensitive.

Note: When a proxy class is specified, WebSphere Portal generates a delegate property. Leave that property the way it is.

7. Restart the portal server for the changes to take effect.

Note: Other than the "BusinessObjects Enterprise XI R2 Portal Integration Kit", Xcelsius Portlets do not need the class loader to be set to PARENT_FIRST for Single Sign-On to work. On the contrary, we strongly discourage changing the Xcelsius Portlet's class loading behavior.

Configuring Single Sign-On

In the Xcelsius Portlet, the Single Sign-On mode is configured by portlet preferences. These preferences may either be set at portlet creation time, by modifying the Xcelsius Portlet's `portlet.xml` file before installation, or in the administration section of the portal.

Parameters that can be modified are the principal's class name, the timeout and the number of logons for which the token should be valid.

Principal class name

The login module appends a so-called Principal to the user's Subject (=an object holding user information and all the user's credentials). In the Xcelsius Portlet, this Principal is needed to create a token. The correct principal is retrieved by its class name. Depending on the login module's authentication mode, the class name varies:

The Principal's Class Name must be one of the following two settings:

- `com.crystaldecisions.sdk.framework.IEnterprisePrincipal` for password authentication
- `com.crystaldecisions.sdk.framework.ITrustedPrincipal` for trusted authentication

The name of the preference is `com.businessobjects.xcelsius.portlet.sso.principalclassname`

Timeout

The timeout property determines the time span for which the token is valid in minutes.

The name of the preference is `com.businessobjects.xcelsius.portlet.sso.timeout`.

Number of Logons

The number of logons property defines for how many logons the token can be used.

The name of the preference is `com.businessobjects.xcelsius.portlet.sso.nooflogons`.

Finding more information

For more information, visit the following web sites:

- WebSphere Portal information and product documentation
" <http://www.ibm.com/software/genservers/portal/support/>"
- Background Information to the Java Authentication and Authorization Service (JAAS)
"
<http://java.sun.com/j2se/1.4.2/docs/guide/security/jaas/JAASRefGuide.html>"
- The BusinessObjects Enterprise Portal Integration Kits:
" <http://www.businessobjects.com/products/downloadcenter/boepik.asp>"
- BusinessObjects product documentation:
" <http://www.businessobjects.com/support>"