My webMethods Server Portlet Reference

Version 10.7

October 2020

WEBMETHODS

This document applies to My webMethods Server 10.7 and to all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: MWS-PORTLET-RF-107-20201015

Table of Contents

About this Guide

This guide provides detailed reference information about the portlets, components, and dynamic business objects (DBOs) that My webMethods Server provides out-of-the-box.

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Narrowfont	Identifies service names and locations in the format *folder.subfolder.service*, APIs, Java classes, methods, properties.
Italic	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Software AG Documentation Website

You can find documentation on the Software AG Documentation website at http://documentation.softwareag.com.

Software AG Empower Product Support Website

If you do not yet have an account for Empower, send an email to empower@softwareag.com with your name, company, and company email address and request an account.

Once you have an account, you can open Support Incidents online via the eService section of Empower at https://empower.softwareag.com/.

You can find product information on the Software AG Empower Product Support website at https://empower.softwareag.com.

To submit feature/enhancement requests, get information about product availability, and download products, go to Products.

To get information about fixes and to read early warnings, technical papers, and knowledge base articles, go to the Knowledge Center.

If you have any questions, you can find a local or toll-free number for your country in our Global Support Contact Directory at https://empower.softwareag.com/public_directory.aspx and give us a call.

Software AG TECHcommunity

You can find documentation and other technical information on the Software AG TECHcommunity website at http://techcommunity.softwareag.com. You can:

- Access product documentation, if you have TECHcommunity credentials. If you do not, you will need to register and specify "Documentation" as an area of interest.
- Access articles, code samples, demos, and tutorials.
- Use the online discussion forums, moderated by Software AG professionals, to ask questions, discuss best practices, and learn how other customers are using Software AG technology.
- Link to external websites that discuss open standards and web technology.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

1 My webMethods Server Administration and Configuration

The chapter describes the portlets and components that My webMethods Server provides out-of-the-box for administering and configuring My webMethods Server.

Summary of Portlets and Components

The following table lists the portlets and components that My webMethods Server provides out-of-the-box for administering and configuring My webMethods Server.

Name	Category and description
Active Session Monitor Portlet	Analysis. Administrators use this portlet to monitor all active My webMethods Server sessions. Administrators can also view the host name and IP address of logged in users and the node in the cluster to which they are logged in.
Cache Administration Portlet	Tools. Administrators use this portlet to clear My webMethods Server caches, for example, the Resource Bundle Cache. An administrator might clear cache if it is not synchronizing properly. The administrator can also use the portlet to rebuild My webMethods Server user interface files, for example, some system JSP pages.
Cache Viewer Portlet	Administration. Administrators use this portlet to monitor the status of My webMethods Server in-memory caches. This information is useful when analyzing or troubleshooting memory footprint and performance issues with the My webMethods Server and Task Engine.
CAF Application Runtime Configuration—Search Bar Portlet	Configuration. Administrators use this portlet to search for CAF applications that are deployed to the current My webMethods Server instance. After locating a CAF application, the administrator can configure the run-time settings of the CAF application.
CAF Application Runtime Configuration—Search Result Portlet	Configuration. Administrators use this portlet to view the results of a search for CAF applications that are deployed to the current My webMethods Server instance. Administrators can configure the run-time settings of the CAF application listed in the search results.
Cluster Admin Portlet	Configuration. Administrators use this portlet to configure, monitor, and control My webMethods Server in either a standalone or clustered deployment.
Content Import/Export Portlet	Administration. Administrators use this portlet to migrate content from one My webMethods Server to another, for example from a staging server to a production server.

Name	Category and description
Content Service Portlet	Content. Administrators use this portlet to manage content services. Content services define the storage locations where content published to My webMethods Server is physically stored.
Database Administration Portlet	Administration. Administrators use this portlet to manage database connections for My webMethods Server.
Email Admin Portlet	Administration. Administrators use this portlet to configure connections for the SMTP server that My webMethods Server uses to send outgoing e-mail messages. Administrators can define one or more SMTP servers. When administrators define multiple SMTP servers, My webMethods Server load balances the traffic among all accessible servers.
Install Administration Portlet	Administration. Administrators use this portlet to install and uninstall My webMethods Server components, that is, portlet applications and web applications. This portlet displays the components that are available for installation, as well as, those already installed so that an administrator can select to uninstall them.
Locale Administration Portlet	User Interface. Administrators use this portlet to configure the default My webMethods Server system locale. Additionally, using the Locale Rules portlet that is embedded in the Locale Administration portlet, administrators can configure locale rules.
Logging Configuration Portlet	Administration. Administrators use this portlet to configure My webMethods Server logging without restarting the My webMethods Server. From this portlet administrators can change the server-side logging categories and specify the threshold of the log messages that My webMethods Server is to log.
Logging Viewer Portlet	Administration. Administrators and page developers use this portlet to view the most recent messages written to the My webMethods Server log files.
Renderer Administration Portlet	Tools. Administrators use this portlet to manage portal renderers. The portlet lists all registered portal renderers and allows administrators to edit them. Additionally, administrators can create new UsePortlet portal renderers.
Rule Administration Portlet	Administration. Administrators use this generic portlet to configure simple rules for any target. For example, My webMethods Server uses specific instances of the Rules Administrationportlet for configuration of Skin and Shell rules.
Scheduled Event Administration Portlet	Administration. Administrators use this portlet to configure and manage existing, recurring events. The recurring events are events that other portlets registered when they were installed.

Name	Category and description
Shell Administration Portlet	User Interface. Administrators use this portlet to create, edit, delete, import, and export shells. Additionally, the administrator uses this portlet to define the system default shell, which is the fallback shell to use when no shell rules are matched.
Shell Editor Portlet	User Interface. Administrators use this portlet to edit the properties of a shell. Administrators can change a shell's name, description, and the parent shell. Additionally, administrators can change individual sections of the shell, such as, the header, footer, left navigation, right navigation, and the title bar.
Skin Administration Portlet	User Interface. Administrators use this portlet to create, edit, delete, import, and export skins. Additionally, the administrator uses this portlet to define the system default skin, which is the fallback skin to use when no skin rules are matched.
Skin Editor Portlet	User Interface. Administrators use this portlet to edit the properties of a skin, specifically the skin's colors, fonts, and images.
System Information Portlet	Configuration. Administrators and developers use this portlet to view general system and session information. The portlet displays My webMethods Server system information and information about the current session in the following categories: Request, Session, and Application.
Update UUID Portlet	Configuration. Administrators use this portlet to enable Universally Unique Identifier (UUID) resolution and to update the UUID attribute value for directory service users in My webMethods Server. In addition, you can use the Cleanup_InvalidUsers utility in this portlet to remove invalid users and memberships of invalid users from roles in My webMethods Server.
Webspaces Administration Portlet	Administration. Administrators use this portlet to set up an unlimited number of webspaces for the My webMethods Server taxonomy. You can define webspaces if you want to provide URLs for My webMethods Server pages that users can bookmark and that are more descriptive.

Active Session Monitor Portlet

Portlet Title	Active Session Monitor
Portlet Name	wm_sessionmonitor
Portlet File Name	wm_sessionmonitor.pdp

Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_sessionmonitor
Default Instances of the portlet	Folders >Administrative Folders >Administration Dashboard >Analysis > Session Monitor

Analysis. Administrators use this portlet to monitor all active My webMethods Server sessions. Administrators can also view the host name and IP address of logged in users and the node in the cluster to which they are logged in.

Cache Administration Portlet

Portlet Title	Cache Administration
Portlet Name	wm_cacheadmin
Portlet File Name	wm_cacheadmin.pdp
Top-level Folder	development
JSR168 Portlet?	No
Alias	/portlet/wm_cacheadmin
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > Cache Administration

Tools. Administrators use this portlet to clear My webMethods Server caches, for example, the Resource Bundle Cache. An administrator might clear cache if it is not synchronizing properly. The administrator can also use the portlet to rebuild My webMethods Server user interface files, for example, some system JSP pages.

The following lists the actions an administrator can perform by using the Cache Administration portlet.

Action	Description
Clear Thing Cache	Flush the metadata object cache.
Clear ACL Cache	Flush the evaluated access rights cache.
Clear Directory Principals Cache	Flush the directory object cache.
Clear Resource Bundle Cache	Flush the resource bundle class cache.
Clear Portlet Content Cache - All	Flush the presentation cache that contains HTML content that old My webMethods Server portlets cached.

Action	Description
Clear Portlet Content Cache - For me	Invalidate all the presentation cache content for the current user.
Clear Portlet Content Cache - For user	Invalidate all the presentation cache content for a specified user.
Clear Portlet Content Cache - For portlet type	Invalidate all the presentation cache content for a specified portlet type.
Clear Portlet Content Cache - For thing instance	Invalidate all the presentation cache content for a specified thing instance (i.e., any portal resource, such as a folder, document, portlet, etc.).
Re-Build JSP Pages - All	Rebuild all deployed JSP/NBSP pages.
Re-Build JSP Pages - For portlet type	Rebuild JSP/NBSP pages for a specified portlet.
Re-Build JSP Pages - For /ui/app directory	Rebuild JSP/NBSP pages under the MWS/server/*instance name*/deploy/portal.war/ui/app directory.
Re-Build JSP Pages - For portlet type	Rebuild JSP/NBSP pages under the MWS/server/*instance name*/deploy/portal.war/ui/include directory.
Re-Build JSP Pages - For portlet type	Rebuild JSP/NBSP pages under the MWS/server/*instance name*/deploy/portal.war/ui/system directory.
Re-Build Skin Stylesheets	Rebuild all deployed skin .csi files.
Re-Build Core JavaScript Files	Does nothing.
Re-Build XType Popup Menus	Rebuild wm_xmenus/menu.js, which is a composite of all xtypes ui/menu.js files.

Cache Viewer Portlet

Portlet Title	Cache Viewer
Portlet Name	wm_cacheview
Portlet File Name	wm_cacheview.pdp
Top-level Folder	development
JSR168 Portlet?	No
Alias	/portlet/wm_cacheview
Default Instances of the portlet	None

Administration. Administrators use this portlet to monitor the status of My webMethods Server in-memory caches. This information is useful when analyzing or troubleshooting memory footprint and performance issues with the My webMethods Server and Task Engine.

The Cache Viewer portlet reports the current information about the state of caches for a specified My webMethods Server instance. If you are using a clustered environment, use the Cache View portlet separately on each individual node in the cluster. The following table lists the type of information that the portlet provides:

Item	Description
Cache size	The current number of entries in the cache
Maximum size	The maximum number of entries allowed in the cache
Valid entries	The number of entries in the cache that are valid
Invalid entries	The number of entries in the cache that are not valid
Expired entries	The number of entries in the cache that have expired
Expirable entries	The number of valid entries in the cache that have not yet expired, which is a number equal to or less than the number of valid entries.

My webMethods Server manages multiple caches. The following caches to not have a direct impact on the performance of My webMethods Server or Task Engine:

- ThingCache
- TaskDataCache
- RoleCache

Normally, you should *not* update the configuration for a cache. However, if you need to reconfigure a cache, for example, to update the resize policy, maximum cache size, or minimum cache size, you can do so by updating the \config\cache.xml configuration file. For detailed information on updating configuration files, refer to *Administering My webMethods Server*.

The "Purge DB Cache" action of the Cache View portlet cleans the database-persistent cache, which you typically do to clear up database space. This database-persistent cache is used by some legacy (non-webMethods CAF) portlets. However, since My webMethods Server version 7.x, few portlets use this database cache.

CAF Application Runtime Configuration—Search Bar Portlet

Portlet Title	Search Bar
Portlet Name	wm_caf_app_runtime_config___searchbar
Portlet File Name	wm_caf_app_runtime_config.war
Top-level Folder	admin

JSR168 Portlet?	Yes
Alias	/portlet/wm_caf_app_runtime_config___searchbar
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > CAF Application Runtime Configuration

Configuration. Administrators use this portlet to search for CAF applications that are deployed to the current My webMethods Server instance. After locating a CAF application, the administrator can configure the run-time settings of the CAF application.

In the My webMethods Server user interface, the CAF Application Runtime Configuration page contains both the Search Bar and Search Result portlets. The Search Bar portlet matches the value an administrator supplies in the search keyword field against CAF application names to find records to display in the search results. To display the results in the Search Result portlet, the Search Bar portlet's Last Search State (lastSearchState) property is wired to the Search Query (queryString) property of the Search Result portlet.

Properties

Initial Search Tab (initialSearchTab)
Identifies the initial tab to display. Specify one of the following:

- `simpleSearch` – Default. Display the portlet with the **Keyword** tab open.
- `savedSearches` – Display the portlet with the **Saved** tab open.
- `savedSearches/details` – Display the portlet with the **Saved** tab open and the saved search details expanded.

Initial Selected Saved Search (initialSelectedSavedSearch)
Identifies the saved search to load when initially displaying the portlet. Specify the name of any saved search that is associated with the portlet instance. If the property has no value, the portlet does not pre-load a saved search.

Run Search On Display (runSearchOnDisplay)
Indicates whether you want the portlet to automatically execute a saved search when it initially displays. Specify one of the following values:

- `true` – Execute the saved search identified by the Initial Selected Saved Search (initialSelectedSavedSearch) property. This simulates the user clicking **Go** in the Search Bar control.
- `false` – Default. Do not execute the saved search.

No Max Results (noMaxResults)
Indicates whether you want to limit the number of the search results to display. Specify one of the following values:

- `true` – Do not limit the search results; that is, display all the results.

- `false` – Default. Limit the search results to the number specified by the Max Results (maxResults) property.

Max Results (maxResults)
Defines the maximum number of search results to display. Specify a positive whole number. The default is 200.

Note:
If the No Max Results (noMaxResults) property is set to `true`, this property is ignored.

Last Search State (lastSearchState)
Facilitates passing the search query value to the Search Result portlet when a user clicks **Go** in the Search Bar control. When a user clicks **Go**, the portlet serializes and converts the active search query object to a base64-encoded string, which is then set as the value of this property. If the property has no value, the portlet behaves as if there is no active search.

Use the My webMethods Server portlet wiring capability to wire the Search Query (queryString) property of the Search Result portlet to this property.

Saved Searches Map (savedSearchMap)
Holds all the private saved searches for the current user. When a user creates and saves a new search, to save the search the portlet adds the search to this property. If the property has no value, no saved searches are available.

CAF Application Runtime Configuration—Search Result Portlet

Portlet Title	Search Result
Portlet Name	wm_caf_app_runtime_config___searchresult
Portlet File Name	wm_caf_app_runtime_config.war
Top-level Folder	admin
JSR168 Portlet?	Yes
Alias	/portlet/wm_caf_app_runtime_config___searchresult
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > CAF Application Runtime Configuration

Configuration. Administrators use this portlet to view the results of a search for CAF applications that are deployed to the current My webMethods Server instance. Administrators can configure the run-time settings of the CAF application listed in the search results.

In the My webMethods Server user interface, the CAF Application Runtime Configuration page contains both the Search Bar and Search Result portlets. The Search Bar portlet finds the CAF applications to display in the search results. To display the results, the Search Bar portlet's Last Search State (lastSearchState) property is wired to the Search Query (queryString) property of the Search Result portlet.

The Search Result portlet displays the search results in a table. Administrators can click the link for a CAF application in the search results to open a user interface that they can use to configure the application.

Properties

Search Query (queryString)
Holds the value of the current search query string. You typically wire this property to the Search Bar portlet's Last Search State (lastSearchState) property. If the property has no value, the portlet displays no results. That is, it behaves as if no search was executed.

Number of Rows to Display (initialPageSize)
Defines the number of rows to display on each page of the search results. Specify a positive whole number. The default is 20.

Sort By (initialSortBy)
Identifies the ID of the column to use to initially sort the search results. Specify one of the following column IDs:

- `applicationColumn` – Sorts the results by the application name.
- `lastModifiedColumn` – Sorts the results by the date the application was last modified.

If the property has no value, the portlet does not initially sort the search results.

Sort Order (initialSortAscending)
Indicates whether to sort the search results in ascending or descending order. Specify one of the following:

- `true` – Default. Sort the search results in ascending order.
- `false` – Sort the search results in descending order.

Column Display (columnDisplay)
Identifies the IDs of all the columns that should be visible in the search results. Specify the column IDs in comma separated values (CSV) format. If the property has no value, the portlet displays all columns.

Cluster Admin Portlet

Portlet Title	Cluster Admin
Portlet Name	wm_mws_config___clusteradmin
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	Yes; however, the portlet is not portable because its functionality is My webMethods Server-specific.
Alias	/portlet/wm_mws_config___clusteradmin

Default Instances of the portlet	■ The system default instance is under folder.portlets.admin folder ■ When logged in as "sysadmin", the default instance is under folder.admin.dashboard.portal_configuration folder ■ Folders > Administrative Folders > Administration ■ Dashboard > Configuration > Cluster Administration ■ When logged in as "Administrator", the default instance is under webm.apps.config.url.setup folder ■ Administration > My webMethods > Cluster Settings

Configuration. Administrators use this portlet to configure, monitor, and control My webMethods Server in either a standalone or clustered deployment.

- **For a standalone deployment**, administrators use the portlet to:
  - Configure HTTP and HTTPS listener ports for My webMethods Server.
  - Configure active server roles for My webMethods Server.
  - Monitor the online status, up time, and active users for My webMethods Server.
  - Shut down and restart My webMethods Server.
- **For a clustered deployment**, administrators use the portlet to:
  - Configure HTTP and HTTPS listener ports for each My webMethods Server node in the cluster.
  - Configure active server roles for each My webMethods Server node in the cluster.
  - Configure the front-end URL for the cluster.
  - Configure the component deployment mode in the clustered environment.
  - Monitor the online status, up time, and number of active users of all My webMethods Server nodes in the cluster.
  - Shut down and restart individual My webMethods Server nodes in the cluster or to shut down and restart the entire cluster.

Content Import/Export Portlet

Portlet Title	Content Import/Export
Portlet Name	wm_migration
Portlet File Name	wm_migration.pdp
Top-level Folder	admin

JSR168 Portlet?	No
Alias	/portlet/wm_migration
Default Instances of the portlet	Administrative Folders > Administration Dashboard > Migration > Content Import/Export

Administration. Administrators use this portlet to migrate content from one My webMethods Server to another, for example from a staging server to a production server.

To migrate data, an administrator uses the Content Import/Export portlet on one server to export content to an external file. The administrator can then import the information from the external file into another My webMethods Server using the Content Import/Export portlet on that server. Administrators can migrate documents, folders, external links, internal links, pages, portlets, Dynamic Business Objects (DBOs), permissions, subscriptions, and portlet wiring properties.

For more information about using the portlet, see information about managing My webMethods Server content in *Administering My webMethods Server*.

Content Service Portlet

Portlet Title	Content Service
Portlet Name	wm_mws_config___contentserviceadmin
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_mws_config___contentserviceadmin
Default Instances of the portlet	Administrative Folders > Administration Dashboard > Content > Content Service

Content. Administrators use this portlet to manage content services. Content services define the storage locations where content published to My webMethods Server is physically stored. This portlet is included within wm_mws_config.war

The physical storage location for a content service is either an FTP server or a location in the file system. If a file system location is used, it can be either a mapped network drive to an external file server or a separate hard drive on the My webMethods Server machine.

An administrators can use this portlet to:

- Create, configure, and delete content services.
- Designate a default content service to define the storage location that My webMethods Server is to use for all new published files.
- Set the maximum size for the uploaded files.

- Import folders and files to create a new content service that has a My webMethods Server folder/file taxonomy that mirrors the source file system. This operation can be performed only once per content service, that is, once per external file system.

Database Administration Portlet

Portlet Title	Database Administration
Portlet Name	wm_mws_config___datasourceadmin
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_mws_config___datasourceadmin
Default Instances of the portlet	■ When logged in as sysadmin: Folders > Administrative Folders > Administration Dashboard > Configuration > DataSource Administration ■ When logged in as Administrator: Administration > My webMethods > Data Sources

Administration. Administrators use this portlet to manage database connections for My webMethods Server. This portlet is included within wm_mws_config.war.

Administrators can create database connections for supported RDBMS (DB2, Oracle, SQL Server, Sybase, MySql), or create custom connections using a bundled DataDirect JDBC driver. For a custom connection the administrator specifies the JDBC driver and JDBC URL. To create a custom connection that does not use a DataDirect JDBC driver, the administrator must add the driver jar to the My webMethods Server run-time classpath.

The portlet contains a 'default' connection for the MWS database schema; administrators cannot modify this connection.

Email Admin Portlet

Portlet Title	Email Admin
Portlet Name	wm_mws_config___emailadmin
Portlet File Name	wm_mws_config___emailadmin.pdp
Top-level Folder	admin
JSR168 Portlet?	Yes
Alias	/portlet/wm_mws_config___emailadmin

Default Instances of the portlet

When logged in as sysadmin, the default instance is:

Folders > Administrative Folders > Administration Dashboard > Configuration > E-mail Administration

When logged in as Administrator, the default instance is:

Administration > My webMethods > E-Mail Servers

Administration. Administrators use this portlet to configure connections for the SMTP server that My webMethods Server uses to send outgoing e-mail messages. Administrators can define one or more SMTP servers. When administrators define multiple SMTP servers, My webMethods Server load balances the traffic among all accessible servers.

E-Mail Properties

Administrators can configure the following properties for an SMTP e-mail server.

Transport Protocol
Identifies the e-mail protocol to be used. The default and only valid value is `smtp`.

SMTP Hosts
Identifies the SMTP server. For example: `smtp.server.com`. If you specify two or more hosts, type one address per line.

SMTP Port
Identifies the port number. Specify the SMTP server's port number, for example, `25`.

SMTP Username
Optional. Identifies the user name that My webMethods Server is to supply for authentication. If the SMTP server requires authentication, specify the user name to supply to satisfy the authentication challenge.

SMTP Password
Optional. Identifies the password associated with the SMTP Username . If the SMTP server requires authentication, specify the appropriate password.

SMTP TLS Enabled
Indicates whether to use an encrypted SMTP connection (by means of TLS). If the SMTP server requires authentication, specify one of the following values:

- `true` – Enable TLS.
- `false` – Default. Do not enable TLS.

SMTP Timeout
Defines the maximum period of time to wait for a response from the server, specified in milliseconds. Default value is 60000.

SMTP Connection Timeout
Defines the maximum period of time for a given SMTP session, specified in milliseconds. Default value is 60000.

SMTP Debug Enabled
Indicates whether to enable debugging for e-mail activities. My webMethods Server writes the debugging information to the My webMethods Server logs. Specify one of the following values:

- `true` – Enable debugging.
- `false` – Default. Do not enable debugging.

From Name
Defines the default "From" name. Specify the default name to use in the "From" field of the e-mail messages that My webMethods Server sends using the SMTP server.

> **Note:**
> Text in this field is subject to the requirements of the RFC822 Internet Text Message standard. For example, text in parentheses, as in "(Important)", is treated as a comment and is removed when the message is created, and bracketed text, such as "[Status]", is treated as an optional element and is also removed.

From E-Mail Address
Defines the default "From" e-mail address. Specify the default e-mail address to use in the "From" field of the e-mail messages that My webMethods Server sends using the SMTP server.

Skin
Identifies the skin to use when rendering My webMethods Server e-mail notifications. Specify a My webMethods Server skin.

Admin Email Address
Defines the email address of the My webMethods Server administrator. This is used as the 'from' address for administrative e-mail messages sent on behalf of the server.

Install Administration Portlet

Portlet Title	Install Administration
Portlet Name	wm_mws_config___installadmin
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	Yes; however, the portlet relies on My webMethods Server internal APIs and will fail to function in other portlet containers.
Alias	/portlet/wm_mws_config___installadmin
Default Instances of the portlet	Administrative Folders > Administration Dashboard > Configuration > Install Administration

Administration. Administrators use this portlet to install and uninstall My webMethods Server components, that is, portlet applications and web applications. This portlet displays the components that are available for installation, as well as, those already installed so that an administrator can select to uninstall them.

The portlet displays the components in a tree structure that reflects the directory structure under the *Software AG_directory* /MWS/components directory. For each component in the tree:

- The **Status** column indicates whether the component is installed, has an update available, or is currently not installed.
- The **Action** column displays the commands available for the component, for example, Install or Upgrade.

The administrator can perform an action on multiple components at one time by first selecting multiple components, and then clicking either **Install Selected** or **Uninstall Selected**.

Administrators can also use this portlet to install components that are not located with in the *Software AG_directory* \MWS\components directory structure, but that are available on the local file system or via a standard URL (e.g., file://, ftp://, http://, etc.). To do so, the administrator can click **Install New Component** to use a wizard for locating and installing the components.

Locale Administration Portlet

Portlet Title	Locale Administration
Portlet Name	wm_localeadmin
Portlet File Name	wm_localeadmin.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_localeadmin
Default Instances of the portlet	Administrative Folders > Administration Dashboard > User Interface > Locale Administration

User Interface. Administrators use this portlet to configure the default My webMethods Server system locale. Additionally, using the Locale Rules portlet that is embedded in the Locale Administration portlet, administrators can configure locale rules.

My webMethods Server uses the system locale and locale rules when determining the language to use during a user session:

- **System locale** is the default language to use when a locale setting is not otherwise defined.
- **Local rules** are evaluated by My webMethods Server to determine a locale.

The following describes how My webMethods Server determines the locale to use for a user session:

1. For a non-guest user, My webMethods Server obtains the locale preference from the User Profile.
2. If the locale preference is not set in the User Profile or the user is a guest, My webMethods Server evaluates locale rules. If one of the rules is triggered, My webMethods Server uses the locale from the rule.

My webMethods Server only evaluates locale rules at login time or when a guest session is first created. As a result, you cannot define locale rules for My webMethods Server resources (pages, folders, portlets).

3. If no locale rule is triggered, My webMethods Server checks the user's web browser language preference.

   My webMethods Server obtains this preference from the Accept-Language request header and uses the language with the highest preference. Typically, browsers have default settings for user accepted languages and locales.

4. If no language preference is defined for the web browser, My webMethods Server uses the default My webMethods Server system locale.

After My webMethods Server sets the initial locale for the user session, the end user can change it by:

1. Using the Locale Switcher Portlet.

   Page developers can place the Locale Switcher portlet on every page by placing it in the portal shell. My webMethods Server uses the new locale until the end user changes it again or until the user session ends.

2. Setting the User Locale field in the User Profile Page. When an end user changes this field in their own profile, the change takes affect immediately. Changes to the User Profile are persisted, and as a result, the change continues to be in effect the next time the end user logs in.

Logging Configuration Portlet

Portlet Title	Logging Configuration
Portlet Name	wm_mws_config___loggingconfiguration
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	Yes
Alias	/portlet/wm_mws_config___loggingconfiguration
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Analysis > Logging Configuration

Administration. Administrators use this portlet to configure My webMethods Server logging without restarting the My webMethods Server. From this portlet administrators can change the server-side logging categories and specify the threshold of the log messages that My webMethods Server is to log.

Logging Threshold Settings of the Logging Configuration Portlet

Administrators use these settings to specify how much information My webMethods Server logs. My webMethods Server logs messages at the level the administrator sets and all higher levels. The logging levels are:

Level	Description
DEBUG	Indicates that the server is to log DEBUG messages and all other log levels. The server issues DEBUG messages at multiple points within a My webMethods Server event. Include DEBUG messages in the log when debugging problems. Note that the log files will grow quickly.
INFO	Indicates that the server is to log INFO messages and all other log levels except DEBUG. The server issues INFO messages when a My webMethods Server event occurs.
WARN	Indicates that the server is to log only WARN and FATAL messages. The server issues WARN messages when a non-fatal error occurs.
FATAL	Indicates that the server it to only log FATAL messages. The server issues FATAL messages when a fatal error occurs.

Configure the logging level for each of the two stages for which My webMethods Server collects logging messages.

- **Category** is the first stage where My webMethods Server collects messages. For each logging category, specify a logging level. When My webMethods Server collects a log message, it sends the message to a specified category. If the level of the message meets or exceeds the logging level configured for the category threshold, My webMethods Server sends the message to the logging outputs. If the message is at a level lower than the configured level for the category threshold, My webMethods Server discards the message.
- **Output** receives messages from the category stage. The My webMethods Server forwards messages that meet a category threshold to valid outputs for the category. If the level of the message meets or exceeds the logging level configured for the output threshold, My webMethods Server writes the message to the output. If the message is at a level lower than the configured level for the output threshold, My webMethods Server discards the message.

A category threshold takes precedence over an output threshold; that is, a logging message that My webMethods Server discards for a category cannot be written to output.

Logging Viewer Portlet

Portlet Title	Logging Viewer
Portlet Name	wm_mws_config___loggingviewer
Portlet File Name	wm_mws_config.war

Top-level Folder	admin
JSR168 Portlet?	Yes
Alias	/portlet/wm_mws_config___loggingviewer
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Analysis > Logging Viewer

Administration. Administrators and page developers use this portlet to view the most recent messages written to the My webMethods Server log files.

Using the portlet, an administrator or page developer identifies the log file to view and the number of lines from that log to display. The portlet displays the last 'n' lines written to the log file, where 'n' is the number of lines the administrator or page developer specify.

Renderer Administration Portlet

Portlet Title	Renderer Administration
Portlet Name	wm_renderadmin
Portlet File Name	wm_renderadmin.pdp
Top-level Folder	development
JSR168 Portlet?	No
Alias	/portlet/wm_renderadmin
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > User Interface > Renderer Administration

Tools. Administrators use this portlet to manage portal renderers. The portlet lists all registered portal renderers and allows administrators to edit them. Additionally, administrators can create new UsePortlet portal renderers.

A portal renderer is a My webMethods Server component that renders My webMethods Server pages. UsePortlet portal renderers render the standard My webMethods Server shell header and footer, but delegate the rendering of the page's main content to another portlet.

- **To create a new UsePortlet renderer**, an administrator selects the **Create New Renderer** tab of the Renderer Administration portlet, selects the portlet that the new renderer is to use for the page's main content, and configures the portlet's properties.

  To use the new renderer for a page, the administrator can add the "layout" parameter to the page's URL. For example, to display the Public Folders page with a renderer named "myRenderer", the following URL might be used:

  ```
  http://localhost:8585/folder.public?layout=myRenderer
  ```

An administrator can also use rendering rules to automatically apply a new renderer to a page; for more information, see *Administering My webMethods Server*.

- **To edit the properties of an existing portal renderer**, an administrator pages through the list of existing renderers to locate the portal renderer to update, and then selects the **Modify Renderer** option of the Tools menu for that renderer. For UsePortlet renderers, an administrator can update any property of the portlet used by the renderer; however, for a non-UsePortlet renderer, an administrator can only edit the renderer's name and description.

  The portlet also allows administrators to edit the raw configuration for all types of portal renderers. To do so, an administrator pages through the list of existing renderers to locate the renderer to update, and then clicks the name of that renderer. In response, the portlet displays the raw properties page that lists the raw XML configuration data that the administrator can edit.

Properties

Cache Age (CacheAge)
Defines how long to cache the contents of the Renderer Administration portlet. Specify one of the following:

- `0` – Default. Do not cache the contents of the portlet.
- `-1` – Cache the contents of the portlet indefinitely.
- *number* – Cache the contents for the specified number of minutes.

All instances of the portlet share the value of this property.

Portlet Properties

Portlet (portlet)
Required. Identifies the portlet that the new renderer is to use to render the page's main content. Specify the thingID or alias of the portlet. If the property has no value, the Renderer Administration portlet will not create the new renderer.

When using the Renderer Administration portlet user interface, this property is automatically set based on the portlet selected on the **Create New Renderer** tab. When programmatically using the Renderer Administration portlet, you must specify a value for this property.

Renderer Properties

Renderer Name (rendererName)
Required. Identifies the name of the renderer that is being created or edited. Specify 1 through 100 characters for the name. If the property has no value, the Renderer Administration portlet will not create or update the renderer.

When using the Renderer Administration portlet user interface, this property is automatically set based on the name you specify for the new renderer on the **Create New Renderer** tab or the name of the renderer you selected for edit. When programmatically using the Renderer Administration portlet, you must specify a value for this property.

Renderer Description (rendererDescription)
Provides a description of the renderer being created or edited. If the property has no value, the renderer will not have a description.

When using the Renderer Administration portlet user interface, this property is automatically set based on the description you specify in the user interface. When programmatically using the Renderer Administration portlet, specify a value for this property if you want to provide a description.

Params Properties

Renderer ID (renderedID)
Required. Identifies the renderer that you want to edit by the thingID or alias of the renderer.

When using the Renderer Administration portlet user interface, this property is automatically set to the ID of the of the UsePortlet renderer. When programmatically using the Renderer Administration portlet, you must specify a value for this property.

Rule Administration Portlet

Portlet Title	Rule Administration
Portlet Name	wm_rule_admin
Portlet File Name	wm_rule_admin.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_ruleadmin
Default Instances of the portlet	The following are instances of the Rule Administration portlet, which are located with in the Administrative Folders > Administration Dashboard > User Interface folder: ■ Manage Locale Rules ■ Manage Login Page Rules ■ Manage Rendering Rules ■ Manage Shell Rules ■ Manage Skin Rules ■ Manage Start Page Rules

Administration. Administrators use this generic portlet to configure simple rules for any target. For example, My webMethods Server uses specific instances of the Rules Administration portlet for configuration of Skin and Shell rules.

The portlet allows administrators to add, update, or delete rules and also to change their evaluation order. When creating and updating a rule, the administrator specifies the conditions to match the rule and the target that the rule should resolve to.

Properties

Rule Folder (ruleFolderURI)
Required. Identifies the folder where you want the configured rules to be stored. Specify the thingID of the target folder.

Rule Text Resource Bundle (ruleTextResourceBundle)
Required. Identifies the Java resource bundle that contains the text strings to display in the portlet user interface. Specify the fully-qualified name of the Java resource bundle.

View Rules Text Key (viewRulesTextKey)
Required. Indicates the text to display at the top of the portlet's **View Rules** tab. Specify the key in the resource bundle that contains the appropriate text.

Eval Order Text Key (evalOrderTextKey)
Required. Indicates the text to display at the top of the portlet's **Change Evaluation Order** tab. Specify the key in the resource bundle that contains the appropriate text.

Add Rule Text Key (addRuleTextKey)
Required. Indicates the text to display at the top of the portlet's **Create New Rule** tab. Specify the key in the resource bundle that contains the appropriate text.

Modify Rule Text Key (modifyRuleTextKey)
Required. Indicates the text to display at the top of the portlet's **Modify Rule** tab. Specify the key in the resource bundle that contains the appropriate text.

Clone Rule Text Key (cloneRuleTextKey)
Required. Indicates the text to display at the top of the portlet's **Copy Rule** tab. Specify the key in the resource bundle that contains the appropriate text.

Rule Editing Context (ruleEditingContextURI)
Required. Identifies a rule editing context portlet that the Rule Administration portlet is to invoke to draw the user interface for editing the specific rule type for the portlet instance. Specify the alias or thingID of the rule editing context portlet.

Scheduled Event Administration Portlet

Portlet Title	Scheduled Event Administration
Portlet Name	wm_mws_config___scheduledeventsadmin
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	Yes; however, the portlet is not portable because its functionality is My webMethods Server-specific.

Alias	/portlet/wm_mws_config___scheduledeventsadmin
Default Instances of the portlet	Administrative Folders > Administration Dashboard > Configuration > Scheduled Event Administration

Administration. Administrators use this portlet to configure and manage existing, recurring events. The recurring events are events that other portlets registered when they were installed.

Using this portlet, administrators (who are logged in as sysadmin) can:

- Change the recurring interval
- Pause and resume the event
- Delete the event

Shell Administration Portlet

Portlet Title	Shell Administration
Portlet Name	wm_shelladmin
Portlet File Name	wm_shelladmin.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_shelladmin
Default Instances of the portlet	Administrative Folders > Administration Dashboard > User Interface > Shell Administration

User Interface. Administrators use this portlet to create, edit, delete, import, and export shells. Additionally, the administrator uses this portlet to define the system default shell, which is the fallback shell to use when no shell rules are matched.

For more information about managing shells, see *Administering My webMethods Server*.

Shell Editor Portlet

Portlet Title	Shell Editor
Portlet Name	wm_shelleditor
Portlet File Name	wm_shelleditor.pdp
Top-level Folder	admin
JSR168 Portlet?	No

Alias	portlet.shelleditor
Default Instances of the portlet	No default instance. However, the Shell Editor Portlet uses this portlet for its edit functionality.

User Interface. Administrators use this portlet to edit the properties of a shell. Administrators can change a shell's name, description, and the parent shell. Additionally, administrators can change individual sections of the shell, such as, the header, footer, left navigation, right navigation, and the title bar.

For more information about modifying shells, see *Administering My webMethods Server*.

Skin Administration Portlet

Portlet Title	Skin Administration
Portlet Name	wm_skinadmin
Portlet File Name	wm_skinadmin.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_skinadmin
Default Instances of the portlet	Administrative Folders > Administration Dashboard > User Interface > Skin Administration

User Interface. Administrators use this portlet to create, edit, delete, import, and export skins. Additionally, the administrator uses this portlet to define the system default skin, which is the fallback skin to use when no skin rules are matched.

For more information about managing skins, see *Administering My webMethods Server*.

Skin Editor Portlet

Portlet Title	Skin Editor
Portlet Name	wm_skineditor
Portlet File Name	wm_skineditor.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	portlet.skineditor
Default Instances of the portlet	No default instance. However, the Skin Editor Portlet uses this portlet for its edit functionality.

User Interface. Administrators use this portlet to edit the properties of a skin, specifically the skin's colors, fonts, and images.

For more information about modifying shells, see *Administering My webMethods Server*.

System Information Portlet

Portlet Title	System Information
Portlet Name	wm_mws_config___portalinfo
Portlet File Name	wm_mws_config.war
Top-level Folder	admin
JSR168 Portlet?	Yes
Alias	/portlet/wm_mws_config___portalinfo
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Analysis > System Information

Configuration. Administrators and developers use this portlet to view general system and session information. The portlet displays My webMethods Server system information and information about the current session in the following categories: Request, Session, and Application.

Update UUID Portlet

Portlet Title	Update UUID
Portlet Name	UserDirectory_UpdateUUID
Portlet File Name	UserDirectory_UpdateUUIDPortlet.war
Top-level Folder	extras
JSR168 Portlet?	Yes
Alias	/portlet/UserDirectory_UpdateUUID
Default Instances of the portlet	Folders > System > Portlets > UserDirectory_UpdateUUIDPortlet

Configuration. Administrators use this portlet to enable Universally Unique Identifier (UUID) resolution and to update the UUID attribute value for directory service users in My webMethods Server. In addition, you can use the Cleanup_InvalidUsers utility in this portlet to remove invalid users and memberships of invalid users from roles in My webMethods Server.

For more information about updating UUID and removing invalid users, see *Administering My webMethods Server*.

Webspaces Administration Portlet

Portlet Title	Webspaces Administration
Portlet Name	wm_webspaceadmin
Portlet File Name	wm_webspaceadmin.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_webspaceadmin
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > Webspaces Administration

Administration. Administrators use this portlet to set up an unlimited number of webspaces for the My webMethods Server taxonomy. You can define webspaces if you want to provide URLs for My webMethods Server pages that users can bookmark and that are more descriptive.

Webspaces provide hierarchical URLs for My webMethods Server pages by using page names such as /folder.public/Samples/PortletControlsSample for a Portlet Controls Sample folder in the Samples folder of Public Folders.

By default, the URLs to My webMethods Server pages do not reflect the location of the page in the MWS taxonomy. For example, a folder called Portlet Controls Sample published in the Samples folder of the Public Folders might have an URL like the following:

http://localhost:8585/meta/default/folder/0000001234

Administrators can define webspaces for portions of the My webMethods Server to include in a page's URL. For example, if an administrator sets up a webspace for Public Folders, which has an alias of "folder.public", then the item called Portlet Controls Sample that is published in the Samples folder of the Public Folders would have the following URL:

http://localhost:8585/folder.public/Samples/PortletControlsSample

This URL shows the location of the Portlet Controls Sample item within the Public Folders webspace, because it shows the path from the webspace root to the item: Public Folders > Samples > Portlet Controls Sample.

To define a new webspace with the Webspaces Administration Portlet, do the following:

1. Click the **New WebSpace** tab in the Webspaces Administration Portlet.
2. Use the MWS folder browser to select a webspace root for the Webspace property, for example, Public Folders.
3. Type an alias into the **Alias Name** field.

> **Note:**
> Public Folders includes a default alias of folder.public, so the field can be left blank.

4. Click **Finish**.

Once you define a new webspace, when you browse to that folder (like Public Folders), all links to items within that folder provide intuitive URLs that show their location in the webspace. For example, if you have a Samples for in Public Folders, the URL for it will now be:

http://localhost:8585/folder.public/Samples

A Portlet Controls Sample item within the Samples folder will have a URL of:

http://localhost:8585/folder.public/Samples/PortletControlsSample

Properties

Cache Age (cacheAge)
Indicates how long, in minutes, to cache content. All instances of the portlet share the value of this property.

The following table lists the values that are available in the user interface for the cache age property:

Value	Description
0	Default. The portlet is not cached.
1	1 minute
10	10 minutes
15	15 minutes
20	20 minutes
30	30 minutes
60	1 hour
-1	forever

2 My webMethods Server Security

This chapter describes the portlets and components that My webMethods Server provides out-of-the-box related to My webMethods Server security.

Summary of Portlets and Components

The following table lists the portlets and components that My webMethods Server provides out-of-the-box related to My webMethods Server security.

Name	Description
Http Header Authentication Admin Portlet	Security. Administrators use this portlet to secure My webMethods Server authentication with an external security provider, for example, SiteMinder or Oblix.
Impersonate User Portlet	Administration. Administrators use this portlet to assume the identity of a different user when diagnosing a problem, to verify the functionality or security of a user account, or to perform an activity on behalf of another user.
Login Portlet	System. End users use this portlet to provide their credentials for My webMethods Server authentication.
Login or Logout Link Portlet	System. End users use this portlet to log in or log out of My webMethods Server. The portlet displays the Login link for guest users, displays the Logout link for users who can logout, and displays nothing for all other users.
NTLM Authentication Administration Portlet	Security. Administrators use this portlet to configure single sign on when using an Active Directory infrastructure. When using NT LAN Manager (NTLM) authentication, users who log into a Windows domain do not have to re-authenticate to log into My webMethods Server. Administrators use the portlet to identify the Primary Domain Controller, which is a Microsoft Windows server that handles all accounts in the domain.
Portal Lockdown Administration Portlet	Security. Administrators use this portlet to secure who has the ability to log into My webMethods Server. Using this portlet, administrators restrict login access to users who have IP addresses within specified ranges, for example, IP address inside the corporate firewall. Configure up to four allowed IP address ranges.
RememberMe Auth Admin Portlet	Security. Administrators use this portlet to configure My webMethods Server to use the RememberMe authorization scheme. When the RememberMe authorization scheme is in use, My webMethods Server displays a check box on the Login page. A user selects the check box to have My webMethods Server save credentials in a RememberMe cookie. In subsequent login requests, My webMethods Server attempts to authenticate the user with credentials from the RememberMe cookie.

Name	Description
SAML Authentication Handler Portlet	Security. Administrators configure this portlet to enable the My webMethods Server's SAML single sign-on feature. This portlet is installed by default.
Security Realms Administration Portlet	Security. Administrators use this portlet to define security realms. Use security realms to simplify permissions management. By default, administrators define permissions for each resource in My webMethods Server. Instead, using this portlet, administrators can define security realms, assign resources to the security realms, and then set permissions for the security realms. As a result, rather than individual permissions per resource, permissions are against a collection of resources in a security realm.
XSRF Security Configuration Portlet	Security. System administrators use this portlet to configure Cross-Site Request Forgery (XSRF) countermeasures for My webMethods Server. To combat XSRF, My webMethods Server requires a special token, called Anti-Cross-Site-Request-Forgery Token (AXSRFT), to be present on HTTP requests that invoke My webMethods Server actions, such as a request to delete a folder or change a user's profile information. My webMethods Server generates a unique token for each user, and periodically updates it, once a day by default. When you use the My webMethods Server user interface to perform actions that require an AXSRFT, the My webMethods Server user interface automatically supplies the correct token, so you do not need to do anything special for these actions when using the My webMethods Server user interface.

Http Header Authentication Admin Portlet

Portlet Title	Http Header Authentication Admin
Portlet Name	wm_httpheaderauth
Portlet File Name	wm_httpheaderauth.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	portlet/wm_httpheaderauth
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > Http Header Authentication Administration

Security. Administrators use this portlet to secure My webMethods Server authentication with an external security provider, for example, SiteMinder or Oblix.

To allow an external security provider to control My webMethods Server authentication, administrators use this portlet to deploy My webMethods Server into an Enterprise Security Infrastructure. As a result, the external security provider passes the user ID information that My webMethods Server is to use for authentication in the HTTP header. For more information, see information about external configuration credentials in *Administering My webMethods Server*.

Properties

User Header Name (userHeaderName)
Identifies the HTTP header field that you want My webMethods Server to examine to determine the identity of the current user making a request. Specify an HTTP header field. The external security system sets the value of this field in the HTTP header. By default, My webMethods Server uses the sm_user field to determine the current user.

Enable HTTP Header Authentication (enableHttpHeaderAuth)
Indicates whether you want My webMethods Server to actively look for the HTTP header field identified by the User Header Name (userHeaderName) property and automatically log in users. Specify one of the following:

- `true` – My webMethods Server uses the User Header Name (userHeaderName) property to automatically log in users.
- `false` – Default. My webMethods Server does not automatically log in users.

Logout URL (logoutURL)
Defines the page to redirect a user to after the user logs off My webMethods Server. Specify the URL of the page. The ability to redirect a user to another URL is dependent on the external security infrastructure.

If the property has no value, although the user is able to log out of My webMethods Server, the security provider does not recognize the attempt to log off. As a result, the next request following the log off attempt passes through the security provider and My webMethods Server automatically logs the user in again.

Impersonate User Portlet

Portlet Title	Impersonate User
Portlet Name	wm_impersonateuser
Portlet File Name	wm_impersonateuser.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	/portlet/wm_impersonateuser
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > User Management

Administration. Administrators use this portlet to assume the identity of a different user when diagnosing a problem, to verify the functionality or security of a user account, or to perform an activity on behalf of another user.

Using this portlet, an administrator instantly assumes the identity of another user, even if the user has never previously logged into the system. After assuming another user's identity, the administrator will only have the exact permissions of the target user for the duration of the session.

Important:
Because the portlet does not require the user's password, for security reasons be sure to allow only administrators access to the Impersonate User portlet to avoid abuse of this portlet.

Login Portlet

Portlet Title	Login
Portlet Name	wm_login
Portlet File Name	wm_login.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.login
Default Instances of the portlet	Folders > Start > Log

System. End users use this portlet to provide their credentials for My webMethods Server authentication.

The Login portlet is used by the My webMethods Server Forms auth-scheme. Administrators configure the portlet to indicate the page to display after the end user is authenticated. Based on the configuration the portlet either:

- Redirects the end user to a custom start page, which is a dynamic start page based upon rules or user's preference.
- Displays the original URL that the end user requested.

Properties

Login Target (loginTarget)
Defines the My webMethods Server page to which to redirect the end user after authentication. If the property has no value, the portlet displays the original URL that the end user requested.

Allow the user request to Override Login Target (allowRequestOverride)
Indicates the page to display after authentication, either the page identified by the Login Target (loginTarget) property or the original URL that the end user requested. Specify one of the following:

- `true` – Override the Login Target (loginTarget) property and use the original URL that the end user requested.
- `false` – Default. Redirect to the page identified by the Login Target (loginTarget) property.

Remember Me Secret (secret)
This property is not used.

Login or Logout Link Portlet

Portlet Title	Login or Logout Link
Portlet Name	wm_loginorout
Portlet File Name	wm_loginorout.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.loginorout
Default Instances of the portlet	■ Folders > System > Shell Sections > Guest Head > Login/Logout ■ Folders > System > Shell Sections > My webMethods Shared Pages Shell Header > Logout ■ Folders > System > Shell Sections > My webMethods Shell Header > Logout ■ Folders > System > Shell Sections > My webMethods Start Page Shell Header > Logout

System. End users use this portlet to log in or log out of My webMethods Server. The portlet displays the Login link for guest users, displays the Logout link for users who can logout, and displays nothing for all other users.

NTLM Authentication Administration Portlet

Portlet Title	NTLM Authentication Administration
Portlet Name	wm_ntlmauthadmin
Portlet File Name	wm_ntlmauthadmin.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_ntlmauthadmin

Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > NTLM Authentication Administration

Security. Administrators use this portlet to configure single sign on when using an Active Directory infrastructure. When using NT LAN Manager (NTLM) authentication, users who log into a Windows domain do not have to re-authenticate to log into My webMethods Server. Administrators use the portlet to identify the Primary Domain Controller, which is a Microsoft Windows server that handles all accounts in the domain.

Properties

Domain Controller Name (domainControllerName)
Identifies the Primary Domain Controller that My webMethods Server is to contact to complete the single sign on operation. Specify the host name of the Primary Domain Controller. If the property has no value, single sign on is disabled; users logging into My webMethods Server will have to enter credentials.

Portal Lockdown Administration Portlet

Portlet Title	Portal Lockdown Administration
Portlet Name	wm_lockdown
Portlet File Name	wm_lockdown.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	/portlet/wm_lockdown
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration

Security. Administrators use this portlet to secure who has the ability to log into My webMethods Server. Using this portlet, administrators restrict login access to users who have IP addresses within specified ranges, for example, IP address inside the corporate firewall. Configure up to four allowed IP address ranges.

My webMethods Server only presents the authentication request when a client's IP address falls within an allowed range. This prevents hackers who can guess valid user names/passwords from logging in because their client IP addresses do not fall within the allowed ranges.

For more information about this portlet, see information about managing My webMethods Server in *Administering My webMethods Server* .

Properties

Is Active (isActive)

Indicates whether My webMethods Server checks client IP addresses to ensure they fall within allowed ranges before requesting authentication. Specify one of the following:

- `true` – Ensure a client's IP address is within allowed ranges before requesting authentication.
- `false` – Default. Request authentication regardless of a client's IP address.

All instances of the portlet share the value of this property.

Error page (errorPage)
Identifies the page to redirect a user to when the authentication request is rejected. Specify the page inside My webMethods Server that you want displayed. All instances of the portlet share the value of this property. If the property has no value, no reason for the rejected authentication request is provided.

E-mail Address for Login Notification (email)
Identifies the administrator to notify when a user authentication request is rejected. Specify the administrator's e-mail address. All instances of the portlet share the value of this property. If the property has no value, no e-mail notification for the rejected authentication request is provided to the administrator.

Start IP Range (startIP) or End IP Range (endIP)
Identifies the first range of IP addresses. My webMethods Server will present the authentication request to users who have IP addresses within this range. For **Start IP Range (startIP)**, specify the start IP address of the first range, for example, 10.10.10.0. For **End IP Range (endIP)**, specify the end IP address of the first range, for example, 10.10.10.255.

All instances of the portlet share the value of these properties. If the properties have no value, no users can log in.

Start IP Range (startIP2) or End IP Range (endIP2)
Identifies the second range of IP addresses. My webMethods Server will present the authentication request to users who have IP addresses within this range. For **Start IP Range (startIP2)**, specify the start IP address of the second range. For **End IP Range (endIP2)**, specify the end IP address of the second range.

All instances of the portlet share the value of these properties. If the properties have no value, the properties are ignored.

Start IP Range (startIP3) or End IP Range (endIP3)
Identifies the third range of IP addresses. My webMethods Server will present the authentication request to users who have IP addresses within this range. For **Start IP Range (startIP3)**, specify the start IP address of the third range. For **End IP Range (endIP3)**, specify the end IP address of the third range.

All instances of the portlet share the value of these properties. If the properties have no value, the properties are ignored.

Start IP Range (startIP4) or End IP Range (endIP4)
Identifies the fourth range of IP addresses. My webMethods Server will present the authentication request to users who have IP addresses within this range. For **Start IP Range (startIP4)**, specify the start IP address of the fourth range. For **End IP Range (endIP4)**, specify the end IP address of the fourth range.

All instances of the portlet share the value of these properties. If the properties have no value, the properties are ignored.

RememberMe Auth Admin Portlet

Portlet Title	RememberMe Auth Admin
Portlet Name	wm_remembermeauth
Portlet File Name	wm_remembermeauth.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	portlet.remembermeauth
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > RememberMe Auth Admin

Security. Administrators use this portlet to configure My webMethods Server to use the RememberMe authorization scheme. When the RememberMe authorization scheme is in use, My webMethods Server displays a check box on the Login page. A user selects the check box to have My webMethods Server save credentials in a RememberMe cookie. In subsequent login requests, My webMethods Server attempts to authenticate the user with credentials from the RememberMe cookie.

Use the RememberMe Auth Admin portlet to configure a secret random string that the portlet uses to encode the credentials stored in the RememberMe cookie and set how long clients are to keep the RememberMe cookie.

To use the RememberMe authorization scheme for My webMethods Server resources, from the RememberMe Auth Admin portlet administrators can execute a special action that converts resources that use a specific authorization scheme (for example, "forms") to another authorization scheme, that is, the RememberMe authorization scheme.

Properties

Secret (secret)
Holds a random string known only to My webMethods Server and that My webMethods Server uses to encode the RememberMe cookie. You do not need to specify a value for this property. My webMethods Server automatically generates the random string when the portlet is installed. If the property has no value, the portlet will not function and will log an exception for every un-authenticated request indicating that no secret has been configured. All instances of the portlet share the value of this property.

Timeout (timeout)
Defines how long clients should store the RememberMe cookie. Specify a number of days. The default is `365`. All instances of the portlet share the value of this property.

SAML Authentication Handler Portlet

Portlet Title	SAML Authentication Handler
Portlet Name	wm_samlauth
Portlet File Name	wm_samlauth.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_samlauth
Default Instances of the portlet	Administrative Folders > Administration Dashboard > Configuration > SAML Authentication Administration

Security. Administrators configure this portlet to enable the My webMethods Server's SAML single sign-on feature. This portlet is installed by default.

When the SAML single sign-on feature is enabled, My webMethods Server can receive SAML 1.1 artifacts and use them for single sign-on. To enable the feature, an administrator configures the endpoint URL that My webMethods Server is to use to validate the SAML artifacts.

When an end user clicks a link in a page from an external server and that link is to a My webMethods Server page, when SAML single sign-on is in use, the link includes a unique single-use artifact. The target My webMethods Server receives the artifact and attempts to validate it by invoking specific web services at the endpoint that the administrator configured on the SAML Authentication Administration page. The web service responds with the user name of the authenticated user who originally clicked the link.

General Properties

Security Provider URI (securityProviderURI)
Identifies the SAML security provider to use when the SAML authentication handler validates a received SAML artifact. Specify the endpoint of the SAML security provider. If the property has no value, the SAML authentication handler is disabled.

Security Realms Administration Portlet

Portlet Title	Security Realms Administration
Portlet Name	wm_securityreamadmin
Portlet File Name	wm_securityreamadmin.pdp
Top-level Folder	admin
JSR168 Portlet?	No

Alias	/portlet/wm_securityrealmadmin
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard > Configuration > Security Realms Administration

Security. Administrators use this portlet to define security realms. Use security realms to simplify permissions management. By default, administrators define permissions for each resource in My webMethods Server. Instead, using this portlet, administrators can define security realms, assign resources to the security realms, and then set permissions for the security realms. As a result, rather than individual permissions per resource, permissions are against a collection of resources in a security realm.

From this portlet, an administrator can:

- Create security realms.
- Delete security realms.
- Manage permissions assigned to security realms.
- Manage the My webMethods Server resources that are assigned to each security realm.

Administrators can organize the security realms by grouping them into folders, called containers.

Optionally, an administrator can configure the security realm to delegate permission decisions to an external security provider.

To configure the security realm to delegate permission decisions to an external security provider, do the following:

1. Click the name of a Security Realm container, for example, Portal Resources.
2. On the **Tools** menu of the security realm you want to modify (for example, the Administrative Commands security realm), select **Modify Security Realm**.
3. For the **Extended Type** field, click **Browse**.
4. Select the security provider you want to use (for example, if you installed the HTTP Header Policy security provider, select form > wm_xt_httpheaderpolicy).
5. Click **Update**.

When you link the security realm to an external security provider, permissions set on the security realm itself are no longer relevant; all permission decisions are delegated to the security provider.

XSRF Security Configuration Portlet

Portlet Title	XSRF Security Configuration
Portlet Name	wm_axsrftconfig

Portlet File Name	wm_axsrftconfig.pdp
Top-level Folder	admin
JSR168 Portlet?	No
Alias	/portlet/wm_axsrftconfig
Default Instances of the portlet	Administration Dashboard > Configuration > XSRF Security Configuration

Security. System administrators use this portlet to configure Cross-Site Request Forgery (XSRF) countermeasures for My webMethods Server. To combat XSRF, My webMethods Server requires a special token, called Anti-Cross-Site-Request-Forgery Token (AXSRFT), to be present on HTTP requests that invoke My webMethods Server actions, such as a request to delete a folder or change a user's profile information. My webMethods Server generates a unique token for each user, and periodically updates it, once a day by default. When you use the My webMethods Server user interface to perform actions that require an AXSRFT, the My webMethods Server user interface automatically supplies the correct token, so you do not need to do anything special for these actions when using the My webMethods Server user interface.

My webMethods Server ensures that an attacker cannot guess a user's AXSRFT by generating the token with a secret that only My webMethods Server knows. This secret is managed automatically by My webMethods Server; it re-generates a new secret periodically and retains a few recent old secrets to validate recent old tokens. You can configure the interval at which My webMethods Server generates new secrets by means of this portlet's "New Secret Interval" property, and you can configure the duration to retain old secrets by means of this portlet's "Oldest Secret" property. The default settings are to generate a new secret every day and to retain an old secret for two days, thus allowing a user who visited a page late one day to use that same page to invoke an action early the next day.

You can also configure a special whitelist of client IP addresses or hostnames from which HTTP requests are never checked for an AXSRFT. While this whitelist disables the My webMethods Server XSRF countermeasures when using the My webMethods Server user interface in a Web browser on those machines, it enables a system administrator to invoke My webMethods Server actions directly from those machines, without using the My webMethods Server user interface. This is useful, for example, if you have a script on a local machine that performs My webMethods Server actions by means of HTTP requests, such as a script that uploads system logs to My webMethods Server every day, or a script that runs performance tests against My webMethods Server. It may also be useful in an emergency if you need to invoke a My webMethods Server action, but cannot access the normal My webMethods Server user interface, for example, if you need to delete a My webMethods Server folder, but some unexpected error rendering the My webMethods Server user interface prevents you from invoking any actions on that folder from the user interface.

Properties

Oldest Secret (oldestSecret)
Required. Specifies the length of time to keep old AXSRFT secrets, in minutes. Keeping recent old secrets after a new AXSRFT secret has been generated allows users to continue to use My

webMethods Server pages that were rendered before the latest AXSRFT secret was generated, so the value of this property should also be longer than the value of the "New Secret Interval" property. The default value is 2820 minutes, a little less than two days.

New Secret Interval (newSecretInterval)

Required. Specifies the interval between which My webMethods Server generates a new AXSRFT secret, in minutes. This secret is used to generate an unguessable anti-XSRF token for each user. The default values is 1440 minutes (1 day).

Whitelist (whitelist)

Provides a comma separated list of client IP addresses, IP address ranges, or hostnames for which to disable the My webMethods Server XSRF countermeasures. Specify an IP address range as the first IP in the range, a dash character, and the last IP in the range, such as "192.168.0.0-192.168.0.255". For example, a whitelist consisting of "localhost,10.1.0.1-10.1.0.4,pc1.private.corp.com" would disable the My webMethods Server XSRF countermeasures for HTTP connections from localhost (the server running My webMethods Server), 10.1.0.1, 10.1.0.2, 10.1.0.3, 10.1.0.4, and pc1.private.corp.com. If a value is not specified, the default value is empty, which means that the My webMethods Server XSRF countermeasures are enabled for all clients.

3 Page and Portlet Development

This chapter describes portlets and components that My webMethods Server provides out-of-the-box for page and portlet development.

Summary of Portlets and Components

The following table lists the portlets and components that My webMethods Server provides out-of-the-box for page and portlet development.

Name	Description
Atom Renderer Component	External. End users and page developers apply the Atom Renderer component to My webMethods Server resources, for example a page, to generate a feed in Atom Syndication format. The syndication feed that the atom renderer generates can be read with a feed reader application.
Basic Breadcrumbs Portlet	System. Page developers use this portlet to add breadcrumbs to a page to show the current page's location within the My webMethods Server taxonomy.
Error Report Portlet	System. End users use this portlet to view an error message. When the error renderer handles a portlet error, it sends the error report to this Error Report portlet. The Error Report portlet displays the error message and includes a **Details** button that the end use can click to show details about an error.
Folder View Portlet	Content Management. End users use this portlet to display the contents of a folder in the My webMethods Server content management system.
Footer Stick	User Interface. Shell developers use this portlet to make a My webMethods Server page use 100% of the web browser window height. Include this portlet into the shell footer section so that the footer "sticks" to the bottom of the web browser window.
Formatted XML Portlet	Drawing. Page developers use this portlet to display formatted XML content on a My webMethods Server page. Configure the portlet to format the XML content either by using simple indentation or by using an XSLT style sheet.
Frame Portlet	Drawing. Page developers and end users use this portlet to add a resizeable container to display a page. This portlet encapsulates a standard HTML <IFrame> tag as a portlet.
HTML Text Portlet	Drawing. Page developers use this portlet to display a fragment of static HTML on a page. When configuring the portlet, the page developer can either specify the HTML text to display or indicate how to retrieve the HTML text from a resource bundle, for example, for language pack localization.

Name	Description
Image Portlet	Drawing. Administrators and end users use this portlet to add an image to a page or workspace to enhance its look-and-feel. Optionally, administrators or end users can link the image to another My webMethods Server resource so that when a user clicks the image, the system displays the linked resource.
Information Message Portlet	System. Page developers use this portlet to display an information, warning, or error messages on a page for the end user. Based on the configured severity, the portlet displays an appropriate icon to lead the message. The portlet displays the message using the CSS class portlet-msg-info.
Inline Content Portlet	Content Management. Page developers use this portlet to display the contents of a published text file as a portlet.
List of Links Portlet	Content Management. Page developers use this portlet to add navigation to other My webMethods Server pages. The portlet displays the children of a specified folder as a list of links. End users click on the links to navigate to other child pages.
Locale Switcher Portlet	Tools. End users use this portlet to temporarily switch their user session locale. The locale change is effective for the duration of the user session only. A page developer uses the portlet's Show List Of property to configure the locales available for selection. This list can be either all possible locales or be limited to those locales for which there are installed language packs.
Main Nav Portlet	Shared. Page developers use this portlet to display a section of the My webMethods Server taxonomy as one or more rows of tabs. It also displays the subsection of the taxonomy for each item that is contained in the first row of tabs as a hover popup-menu, that is, if the Show Menus (showMenus) property is set to true. The section displayed starts with the My webMethods Server folders configured with the Roots (roots) property and includes descendants of those roots along the path to the My webMethods Server item configured with the Leaf (leaf) property.
Menubar Portlet	Page Components. Page developers add this portlet to a shell or page to provide navigation to other My webMethods Server pages.
Mime Message Folder Renderer Portlet	Email. Page developers use this portlet in My webMethods Server as the default renderer for folder instances that use the data types wm_xt_nntpfolder and wm_xt_pop3folder. This portlet displays the list of e-mail messages contained in the wm_xt_nntpfolder or wm_xt_pop3folder folders.
Mime Message Renderer Portlet	Email. Page developers use this portlet in My webMethods Server as the default renderer for message instances that use the data

Name	Description
	type wm_xt_mimemessage. This portlet displays the content of an e-mail message.
NNTP Folder Threaded View Portlet	Containers. Administrators publish an instance of this portlet to a My webMethods Server page to provide a way to view the contents of an NNTP folder (wm_xt_nntpfolder) as a list of threaded conversations. The administrator configures the portlet to reference an NNTP folder that already exists in the My webMethods Server taxonomy.
Pearls Skin	Skins. This component is a skin. Page developers can use this skin as the default skin or the target of a Skin Rule. Additionally, page developers can use the skin as the base for a custom skin.
Portlet-in-Portlet Portlet	Page Components. A page developer adds this portlet to a page to display the contents of another page within the portlet.
Preformatted Text Portlet	Drawing. Page developers use this portlet to display text content using plain-text formatting.
Publish Portlet	Page Authoring. Page developers and end users use this portlet to publish new instances of My webMethods Server objects. The base types of the My webMethods Server objects that can be published are files, folders, form, links, and portlets.
PublishContext Portlet	Content Management. Page developers use this portlet to simplify the publish process for end users. The PublishContext portlet is a helper for the Publish Portlet, which is a wizard for adding new content to the My webMethods Server content repository. Use this PublishContext portlet to render a button, link, or image control on a page so that when a user clicks the control, the application opens a page that contains the Simple Publish Portlet with the fields of the wizard pre-set. The PublishContext portlet allows you to pre-populate field values, make fields read-only, and/or hide fields in the Simple Publish Portlet wizard.
Quick Access URL Portlet	Content Management. Page developers use this portlet to display an intuitive URL for the page. This intuitive URL is built from the shortest alias to the page. If the page has no alias, this portlet displays nothing.
Renderer Tabs Portlet	Content Management. End users use this portlet to instantly switch the renderer for the current resource (object) being browsed to another renderer. The intended use of the portlet is to include it as part of the shell.
Session History Portlet	Page Components. Page developers add this portlet to a portal shell to provide end users with links to pages they have

Name	Description
	previously visited during their login session. End users click links to navigate back to pages in their history.
Simple Publish Portlet	Page Authoring. Page developers and end users use this portlet to publish new instances of links, documents, folders, and pages.
Simple View Portlet	Content Management. Page developers use this portlet to display the name of and/or link to another item in the My webMethods Server content repository.
Streaming Content Viewer Portlet	System. Page developers publish this portlet to a My webMethods Server page to display a flash widget or streaming media. After publishing the portlet to a page, the page developer configures it to point to a media file that is published elsewhere in My webMethods Server. Based on the file type of the media file, the Streaming Content Viewer portlet uses either the Flash player or Windows Media player to display the media.
Surrogate Portlet	Content Management. Page developers use this portlet to render the user interface of a portlet that resides in another My webMethods Server page. This portlet provides a simple way to share a portlet instance among two or more pages.
Tabs View Portlet	Page Components. Page developers use this portlet to display the contents of a My webMethods Server folder. The portlet displays each item within the folder as a tab. Additionally, the portlet displays the contents of the item associated with the selected tab.
Thing Navigate Up or Down Button Portlet	Title Bar Tools. Page developers use this portlet as part of a shell title bar to provide a way to navigate to the My webMethods Server resource parent folder or back to the current resource.
Thing Popup Menu Portlet	Title Bar Tools. Page developers can use this portlet in custom title bars to provide a default popup menu for a My webMethods Server object. This portlet is part of the core system components and end users should not directly access it.
Thing Portlet Mode Buttons Portlet	Title Bar Tools. Page developers can use this portlet in custom title bars to add icons that allow end users to switch the portlet mode. The supported portlet modes are: view (default), edit, help.
Thing Window State Buttons Portlet	Title Bar Tools. Page developers can use this portlet in custom title bars to add icons that allow end users to switch portlet window states. The supported portlet window states are: normal (default), minimized, maximized.

Name	Description
Time Zone Portlet	Page Components. End users use this portlet to determine the time zone of the displayed page.
Titlebar Guest Titlebar Portlet	Title Bar Tools. Page developers can use this portlet when implementing a custom shell to provide a simple title bar that displays only the portlet title without any actions or menus.

Atom Renderer Component

Component Name	wm_atomrenderer
Component File Name	wm_atomrenderer.cdp
Top-level Folder	default

External. End users and page developers apply the Atom Renderer component to My webMethods Server resources, for example a page, to generate a feed in Atom Syndication format. The syndication feed that the atom renderer generates can be read with a feed reader application.

To apply the atom renderer to a page (or other My webMethods Server content), an end user or page developer creates a link. When a user navigates to the link or if a feed reader is configured with the link, My webMethods Server generates the feed for the page in Atom Syndication format. An easy way to apply the atom renderer to a page is by adding the `layout=atom` parameter to the page URL, for example:

http://localhost:8585/user.current.home?layout=atom

Note:
For descriptions of the link parameters you can specify, see Link Parameters, below.

When a user attempts to use a feed reader to subscribe to the syndication feed, the feed reader must be able to access My webMethods Server. As a result, typically a user must use a desktop-based feed reader. However, a Web-based feed reader can be used if My webMethods Server is accessible outside the corporate firewall or if the Web-based feed reader is hosted within the same firewall as My webMethods Server.

The feed reader must also supply valid My webMethods Server credentials to obtain the feed. This can be done by adding the **username** and **password** link parameters to the feed URL. For example:

http://localhost:8585/user.current.home?layout=atom&username=alice&password=PswD

The user interface of some feed readers includes fields for user authentication. To enable the feed reader to authenticate with credentials from its user interface, the My webMethods Server system administrator must set the authentication scheme for the page (or other My webMethods Server content) for which the feed is generated to use the Basic authentication scheme. The system administrator sets the authentication scheme by using the permissions page. For more information

about authentication, see information about managing security in *Administering My webMethods Server*.

The Atom Renderer component also creates a resource in the Administrative Folders > Global Data folder that is named **Atom Feed Target**, with the alias "atom". The **Atom Feed Target** resource is automatically configured with the basic authentication scheme. An end user or page developer can create a link that applies the atom renderer to the **Atom Feed Target** resource, but specifies a different My webMethods Server resource for which to generate a feed. For example, this link will generate the atom feed for the current user's home folder with the **Atom Feed Target**: "http://localhost:8585/atom?uri=user.current.home".

Link Parameters

The following describes the link parameters that an end user or page developer can specify when creating the link that applies this renderer to a page or other My webMethods Server content.

username
Identifies the My webMethods Server user attempting to subscribe to the atom renderer component. Specify the user's user name. You can omit the **username** and **password** link parameters if the user can provide credentials via the user interface of the feed reader.

password
Identifies the user's password. Specify the password that corresponds to the user name specified by the **username** parameter.

count
Defines the maximum number of entries to generate. The default is `-1`, which indicates there is no maximum. To define a maximum, specify a positive whole number.

startIndex
Identifies the index of the first entry to generate. The default is 0, which indicates the first entry.

For example, if the Atom Renderer component could generate 100 entries for a page, but the link to the feed includes the **count** parameter set to `10` and the **startIndex** parameter set to `20`, the Atom Renderer component would generate a feed with entries 20 through 29.

sort
Identifies the field to use for sorting. Specify the field of the My webMethods Server resource on which to sort the generated entries. By default the Atom Renderer component uses `lastModifiedDate-`, which indicates to sort by the last modified date in descending order.

For example, if the **sort** parameter is set to `name+`, the Atom Renderer component sorts the generated entries by the name field in ascending alphabetical order.

Basic Breadcrumbs Portlet

Portlet Title	Basic Breadcrumbs
Portlet Name	wm_basicbreadcrumbs

Portlet File Name	wm_basicbreadcrumbs.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	/portlet/wm_basicbreadcrumbs
Default Instances of the portlet	None

System. Page developers use this portlet to add breadcrumbs to a page to show the current page's location within the My webMethods Server taxonomy.

Primarily page developers add the portlet as a Shell component. For example, the Shell for the Administration Dashboard uses the Basic Breadcrumbs portlet, and it displays:

Folders > Administrative Folders> Administration Dashboard

Page developers can configure the portlet so that end users can click the breadcrumb nodes to navigate to another folder or page.

Properties

Breadcrumbs root (root)
Identifies the top-level folder for the breadcrumbs. Specify the name of a folder. By default, the portlet uses `Folders`.

For example, if this property is set to `Public Folders`, when an end user displays a page within the Public Folders taxonomy, the Basic Breadcrumbs portlet displays:

Public Folders > My Folder

If this property is not specified and the default, `Folders`, is used, the Basic Breadcrumbs portlet for a page within the Public Folder taxonomy displays:

Folders > Public Folders > My Folder

Breadcrumbs leaf (leaf)
Identifies the expression or location for the right-most breadcrumb node. Typical expressions are:

- `current.resource` – Default. Make the right-most breadcrumb node the current page.
- `resource.parent` – Make the right-most breadcrumb node the parent folder of the current page.

Link Expression (link)
Indicates whether you want end users to be able to navigate to pages or folders by clicking the breadcrumb nodes. Settings are:

- `{uri}` – Default. Make the breadcrumb nodes clickable links so that the end user can navigate to the folder or page specified by the breadcrumb.

- `empty` – Do not make the breadcrumb nodes clickable links.

Exclude Root (exclusive)
Required. Indicates whether the breadcrumb nodes should include the root folder. Specify one of the following:

- `true` – Exclude the root folder from the breadcrumb nodes.
- `false` – Default. Include the root folder in the breadcrumb nodes.

Error Report Portlet

Portlet Title	Error Report
Portlet Name	wm_errorreport
Portlet File Name	wm_errorreport.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.errorReport
Default Instances of the portlet	Folders > Administrative Folders > Portal Error Page

System. End users use this portlet to view an error message. When the error renderer handles a portlet error, it sends the error report to this Error Report portlet. The Error Report portlet displays the error message and includes a **Details** button that the end use can click to show details about an error.

Folder View Portlet

Portlet Title	Folder View
Portlet Name	wm_folderview
Portlet File Name	wm_folderview.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.folderview
Default Instances of the portlet	None

Content Management. End users use this portlet to display the contents of a folder in the My webMethods Server content management system.

Properties

View Depth (viewDepth)
Defines the depth of the descendants to display for a folder. Specify one of the following:

- `1` – Default. Show only the direct children.
- `2` – Show the direct children and their children.
- `3` – Show the direct children, their children, and the children of their children.

Visual Style (visualStyle)
Defines the display mode to use to display the folder contents. Specify one of the following:

- `1` – Default. Display the folder contents as a standard list, a list of items that includes a description and last-modified date.
- `2` – Display the folder contents as a simple list, a condensed list of items that omits the description and last-modified date.
- `3` – Display the folder contents as thumbnail images. Only image files will display a thumbnail. A placeholder image appears for items that do not have a thumbnail, for example, folders, documents, links, and so on.

Show Titlebar (showTitlebar)
Indicates whether to show a title header. Specify one of the following:

- `true` – Default. Show a title header.
- `false` – Hide the title header.

Sort Key (sort)
Defines the key to use for sorting the folder contents. Specify one of the following:

- `name` – Default. Sort by the name of the items in the folder.
- `type` – Sort by the object type of the items in the folder.
- `modified` – Sort by the last modified date of the items in the folder.

Sort Order (order)
Defines the order in which to sort the folder contents. Specify one of the following:

- `ascending` – Default. Sort the folder contents in ascending order.
- `descending` – Sort the folder contents in descending order.

Page Limit (pageSize)
Defines the number of folder items to display on a page. If there are more folder items to display than the limit for the page, the portlet renders paging links so that end users can navigate through all available folder items. Specify one of the following:

- `0` – Default. The portlet uses the **Items Per Page** value from the current user's User Profile.
- `10` – The portlet displays 10 folder items per page.

- `20` – The portlet displays 20 folder items per page.
- `50` – The portlet displays 50 folder items per page.
- `100` – The portlet displays 100 folder items per page.

Folder (listViewTargetURI)
Identifies the folder to display. Specify the thingID or alias of the folder. If the property has no value, the portlet displays the items in the current resource.

Links/Content/Forms Open in New Window (newWindow)
Indicates whether to open a new window when an end user clicks a link in the folder. Specify one of the following:

- `true` – Default. Open the link in a new window.
- `false` – Display the link in the current window.

Reuse Opened Window (reuseWindow)
Indicates whether to reuse the new window opened for a link in the folder when an end user clicks a second link in the folder. Specify one of the following:

- `true` – The portlet opens each linked item in the same new window.
- `false` – Default. The portlet opens each linked item in a unique new window.

This property is ignored if the Links/Content/Forms Open in New Window (newWindow) property is set to `false`. That is, if the link in a folder opens in the current window, all links on subsequent pages also open in the current window.

Footer Stick

Portlet Title	Footer Stick
Portlet Name	wm_footerstick
Portlet File Name	wm_footerstick.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	/portlet/wm_footerstick
Default Instances of the portlet	None

User Interface. Shell developers use this portlet to make a My webMethods Server page use 100% of the web browser window height. Include this portlet into the shell footer section so that the footer "sticks" to the bottom of the web browser window.

Formatted XML Portlet

Portlet Title	Formatted XML
Portlet Name	wm_formattedxml
Portlet File Name	wm_formattedxml.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.formattedxml
Default Instances of the portlet	None

Drawing. Page developers use this portlet to display formatted XML content on a My webMethods Server page. Configure the portlet to format the XML content either by using simple indentation or by using an XSLT style sheet.

Properties

Text (text)
Holds the XML content to display in the portlet. If the property has no value, the portlet displays nothing.

Width (width)
Defines a fixed width for the panel that displays the XML content. Specify a valid CSS width measurement, for example, `350px`, `30em`, or `50%`. If the XML content requires more horizontal space than the panel allows, the portlet either wraps long text or displays a horizontal scroll bar; for more information, see the Word Wrap (wordwrap) property. If the property has no value, the panel uses 100 percent of the available width.

Height (height)
Defines a fixed height for the panel that displays the XML content. Specify a valid CSS height measurement, for example, `350px` or `30em`. If the XML content requires more vertical space than the panel allows, the portlet displays a vertical scroll bar. If the property has no value, the panel grows vertically to show the entire contents without a vertical scroll bar.

Word Wrap (wordwrap)
Indicates whether to wrap long lines or to display a horizontal scroll bar. Specify one of the following:

- `true` – Default. Wrap long lines to the next line if there is not enough horizontal space.
- `false` – Display a horizontal scroll bar if there is not enough horizontal space.

Monospace (monospace)
Indicates whether to use a monospace font to display the XML content. Specify one of the following:

- `true` – Use a monospace font.
- `false` – Default. Do not use a monospace font.

XSL Stylesheet (stylesheet)
Identifies the XSLT style sheet the portlet is to apply to format the XML content. Specify the thingID or alias of an XSLT document that is stored in the My webMethods Server content repository. If the property has no value, the portlet uses simple indentation-based formatting for the XML content.

Cache Content (cacheAge)
Indicates whether to cache the formatted XML content. Caching the XML content speeds up subsequent renderings of the XML content. Specify one of the following:

- `-1` – Cache the XML content.
- `0` – Default. Do not cache the XML content.

Frame Portlet

Portlet Title	Frame
Portlet Name	wm_frame
Portlet File Name	wm_frame.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Drawing. Page developers and end users use this portlet to add a resizeable container to display a page. This portlet encapsulates a standard HTML <IFrame> tag as a portlet.

Properties

URL (URL)
Identifies the page to display in the frame. Specify the URL of the page. The URL can be for a My webMethods Server object or an external page, for example, `www.google.com`. The default value is `about:blank`, which displays an empty page in the frame.

Frame Height (height)
Defines the height of the frame. Specify a value using standard HTML units, for example, `100%` or `250px`. The default value is `100px` to create a frame with the vertical height of 100 pixels.

Frame Name (frameName)

Defines the name property of the IFrame tag. If the property has no value, the frame will not have a name. Specify a name for the frame if you want other HTML components to be able to refer to the frame by name.

HTML Text Portlet

Portlet Title	HTML Text
Portlet Name	wm_htmltext
Portlet File Name	wm_htmltext.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.htmltext
Default Instances of the portlet	None

Drawing. Page developers use this portlet to display a fragment of static HTML on a page. When configuring the portlet, the page developer can either specify the HTML text to display or indicate how to retrieve the HTML text from a resource bundle, for example, for language pack localization.

Writable Text Properties

Text (HTMLText)
Holds the HTML text to display. Specify the static HTML that you want the portlet to display. When you specify a value for **Text (HTMLText)**, the portlet ignores values in the Resource Bundle Class Name (resourceBundleClass) and Resource Bundle Key (resourceBundleKey) properties.

If neither this property nor the Resource Bundle Class Name (resourceBundleClass) and Resource Bundle Key (resourceBundleKey) have values, the portlet displays nothing.

Resource Message Arguments (resourceMsgArgs)
This property specifies a comma-separated value (CSV) to be used in the referenced resource string. For example:

```
Copyright {0} - {1}
```

where the arguments are `2006,2013`. This follows the same Javadoc as String.format().

Read-only Localizable Text Properties

Resource Bundle Class Name (resourceBundleClass)
Identifies the resource bundle that contains the HTML to display. Specify the name of the resource bundle. When you specify this property, you must also specify a value for the Resource Bundle Key (resourceBundleKey) property.

The portlet ignores this property if you specify a value for the Text (HTMLText) property. If you do not specify a value for this property, the portlet either displays the HTML defined by the Text (HTMLText) property or nothing.

Resource Bundle Key (resourceBundleKey)
Identifies the HTML text within the resource bundle that you want the portlet to display. Specify the key of the HTML text. When you specify this property, you must also specify a value for the Resource Bundle Class Name (resourceBundleClass) property.

The portlet ignores this property if you specify a value for the Text (HTMLText) property. If you do not specify a value for this property, the portlet either displays the HTML defined by the Text (HTMLText) property or nothing.

Image Portlet

Portlet Title	Image
Portlet Name	wm_image
Portlet File Name	wm_image.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Drawing. Administrators and end users use this portlet to add an image to a page or workspace to enhance its look-and-feel. Optionally, administrators or end users can link the image to another My webMethods Server resource so that when a user clicks the image, the system displays the linked resource.

General Properties

Name (name)
Required. Identifies the Image portlet being added to the page. Specify the name you want to assign to the Image portlet.

Description (description)
Describes the Image portlet being added to the page. Specify a description. If the property has no value, the Image portlet will not have a description.

Properties

Source (source)
Required. Identifies the image to display. Specify one of the following to identify the source of the image:

- Fully-qualified URL of the image (http, ftp, etc.)
- ThingID or alias of the My webMethods Server content object that is a GIF, PNG, or JPG image file that has been published into My webMethods Server
- Path to an image from a My webMethods Server skin

  My webMethods Server skins define images. Refer to an image from the default skin using the following, where *imagename* is the name of the image file defined in the skin:

  `images/`*imagename*

Width (width)
Defines the width definition for the image. Specify either using the number of pixels (e.g., `100 px`) or the percent of the original size of the image (e.g., `80%`). If the property has no value, the portlet displays the image using its original dimensions.

Height (height)
Defines the height definition for the image. Specify either using the number of pixels (e.g., `100 px`) or the percent of the of the original size of the image (e.g., `80%`). If the property has no value, the portlet displays the image using its original dimensions.

Alignment (align)
Identifies the alignment for the image. Specify left, right, center, top, middle, or bottom. If the property has no value, the image displays middle-aligned.

Link To (link)
Identifies whether you want a user to be able to click the image to display another My webMethods Server resource. To have the image use a link, specify the My webMethods Server resource to which to link. If the property has no value, the image will not use a link.

Pseudo portlet overrides (proxyType)
Identifies the behavior of the Image portlet when it is used as a pseudo portlet for other My webMethods Server objects. This advanced property is only for use by administrators. Specify one of the following:

- `Link (link)` – The image specified for this portlet's source property is displayed, but the image is linked to the other My webMethods Server object.
- `Source (image)` – This portlet displays the content of the other My webMethods Server object as an image if the other My webMethods Server object is a image file, but links to the resource specified by this portlet's link property.
- `Both Link and Source (both)` – Default. This portlet displays the content of the other My webMethods Server object as an image if the other My webMethods Server object is an image file, and the image is linked to the other My webMethods Server object.

Information Message Portlet

Portlet Title	Information Message
Portlet Name	wm_infomessage

Portlet File Name	wm_infomessage.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

System. Page developers use this portlet to display an information, warning, or error messages on a page for the end user. Based on the configured severity, the portlet displays an appropriate icon to lead the message. The portlet displays the message using the CSS class portlet-msg-info.

Properties

Information Text (infoText)
Defines the message to display in the body if the portlet. Specify a text string. If the property has no value, the portlet displays a blank message.

Severity (Severity)
Indicates the icon to display with the message. Specify one of the following:

- `info` – Default. Display the information messages icon.
- `warning` – Display the warning messages icon.
- `error` – Display the error messages icon.

Escape Markup (escapeMarkup)
Indicates whether you want HTML characters in the message be escaped before passing them for display in the message. Specify one of the following:

- `true` – Escape any HTML characters for display in the message, for example, change '\>' to '>'.
- `false` – Default. Pass the text unmodified to the portlet.

Inline Content Portlet

Portlet Title	Inline Content
Portlet Name	wm_inlinecontent
Portlet File Name	wm_inlinecontent.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.inlinecontent

Default Instances of the portlet	None

Content Management. Page developers use this portlet to display the contents of a published text file as a portlet.

Special Media-Type Handling Properties

Plain Text Media-Types (textTypes)
Defines the MIME types that the portlet is to treat as plain text documents. Specify the MIME types using comma-separated values (CSV) format. The default value is:

```
text/*
```

Plain Text Display (text)
Defines how the portlet is to render the contents of documents that have a MIME type identified by the Plain Text Media-Types (textTypes) property. Specify one of the following:

- `link` – Renders a Download link for the plain text content.
- `pre` – Renders the plain text content as preformatted text.
- `soft` – Default. Renders the plain text content as Soft-wrapped text.
- `raw` – Renders the plain text as is, that is, without formatting. The content is treated as if the content is HTML.

HTML Media-Types (htmlTypes)
Defines the MIME types that the portlet is to treat as HTML documents. Specify the MIME types using comma-separated values (CSV) format. The default value is:

```
text/html,application/xhtml*
```

HTML Display (html)
Defines how the portlet is to render the contents of documents that have a MIME type identified by the HTML Media-Types (htmlTypes) property. Specify one of the following:

- `link` – Renders a Download link for the HTML content.
- `pre` – Renders the HTML content as preformatted text.
- `soft` – Renders the HTML content as Soft-wrapped text.
- `raw` – Default. Renders the HTML content as is, that is, without formatting.
- `html` – Renders the HTML content as Script-safe HTML.

XML Media-Types (xmlTypes)
Defines the MIME types that the portlet is to treat as XML documents. Specify the MIME types using comma-separated values (CSV) format. The default value is:

```
text/xml,text/xsl,application/xml*,application/mathml*,application/rdf*,
application/xsl*
```

XML Display (xml)
Defines how the portlet is to render the contents of documents that have a MIME type identified by the XML Media-Types (xmlTypes) property. Specify one of the following:

- `link` – Renders a Download link for the XML content.
- `pre` – Renders the XML content as preformatted text.
- `soft` – Default. Renders the XML content as Soft-wrapped text.
- `tidy` – Renders the XML content as Tidied-XML text.
- `raw` – Renders the XML content as HTML.
- `xsl` – Applies the XSLT style sheet identified by the XML Stylesheet (xmlStylesheet) property and renders the result.

XML Stylesheet (xmlStylesheet)
Identifies the XSLT style sheet that the portlet is to use to format XML content. The portlet only uses this property when the XML Display (xml) property is set to `xsl`. Specify the thingID or alias of an XSLT document that is stored in the My webMethods Server content repository. If the property has no value, the portlet behaves as if the XML Display (xml) property is set to `link`.

Properties

File (proxyThingID)
Identifies the file to display. Specify the thingID or alias of a text document stored in the My webMethods Server content repository. If the property has no value, the portlet uses the current resource.

Display Header (header)
Indicates whether the portlet is to display a simple header above the file content. Specify one of the following:

- `true` – Display a header.
- `False.` – Default. Do not display a header.

Truncate After X Characters (truncate)
Defines the maximum number of characters to display. The portlet displays the number of characters you specify and truncates the rest of the file. Specify a positive whole number or `-1` if you want the portlet to display the contents of the entire file without truncation. The default is `1000` characters.

Cache Content (cacheAge)
Indicates whether to cache the formatted content. Caching the content speeds up subsequent renderings of the content. Specify one of the following:

- `-1` – Cache the content.
- `0` – Default. Do not cache the content.

List of Links Portlet

Portlet Title	List of Links
Portlet Name	wm_listoflinks
Portlet File Name	wm_listoflinks.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.listoflinks
Default Instances of the portlet	None

Content Management. Page developers use this portlet to add navigation to other My webMethods Server pages. The portlet displays the children of a specified folder as a list of links. End users click on the links to navigate to other child pages.

The page developer configures the portlet to specify how to display the links. For example, the page developer can specify the style to use for the links, how to sort the links, how many links to display per page, and whether to only display links that are tagged with a specified keyword.

Properties

Cache Content (cacheAge)
Defines how long to cache the contents of the portlet. Specify one of the following:

- `-1` – Indicates that you want to cache the contents of the portlet indefinitely.
- `0` – Default. Indicates that you do not want to cache the contents of the portlet.
- *minutes* – Defines the number of minutes to cache the contents of the portlet.

All instances of the portlet share the value of this property.

Visual Style (visualStyle)
Defines the style to use for displaying the links. Specify one of the following:

- `1` – Default. Display as tabs.
- `2` – Display as a list of horizontal links.
- `3` – Display as a list of vertical links.
- `4` – Display as subtabs.
- `5` – Display as a drop-down list.

Links Container (rootFolder)

Identifies the folder for which you want to display links. Specify the thingID or alias of the folder. If the property has no value, the portlet uses the parent of the current page.

Selected Link (selectedItem)
Identifies the selected link. Specify the ID of the link that you want the portlet to display as the selected link. If the property has no value, the portlet uses the current page for the selected link.

Force Display of Selected Link (showSelectedItem)
Indicates whether to always show the selected link. This property is used when the XML Display (xml) property identifies a folder that contains more children than the Max items to display (pageSize) property allows on a single page. Specify one of the following:

- `true` – Always display the selected link regardless of the paging state.
- `false` – Default. Only display the selected link when it is in the current paging state.

Link Types (linkType)
Defines the types of objects to display as links. The portlet only displays links for children that are of the type specified. Specify one of the following:

- `basic` – 5
- `container` – Display only children that are of the type folder as links.
- `content` – Display only children that are of the type document as links.
- `link` – Display only children that are of the type links as links.

Max items to display (pageSize)
Defines the maximum number of links to display on a single page. Specify a whole number greater than zero. The default is `10`.

Sort Order (sort)
Identifies how to sort the links. Specify one of the following sorting keys:

- `name` – Default. Sort based on the name of the items.
- `type` – Sort based on the type of items.
- `modified` – Sort based on the last modified date of the items.
- `sortid` – Sort based on the sort order configured for each item. For example, the sortid can be an arbitrary integer for each item in a folder, in a nonconsecutive order, to produce a custom sort order. Items with a respective sortid of 1, 12, 15, and 10 would be sorted in the order of 1, 10, 12, and 15.

Truncate Names After X Characters (truncate)
Defines the maximum length of characters to use for a link name, excess characters will be truncated. Specify a whole number greater than zero. The default is `50`.

Open in New Window (newWindow)
Indicates whether to open links in a new window. Specify on o the following:

- `true` – Default. Open the link in a new window.

- `false` – Display the link in the current window.

Reuse Opened Window (reuseWindow)
Indicates whether to reuse the new window opened for a link when an end user clicks one of the links that this portlet displays. Specify one of the following:

- `true` – If the user clicked another link displayed by this portlet, and the window for that link is open, this portlet will open subsequent links in that same window.
- `false` – Default. The portlet always opens a new window for each link.

This property is ignored if the Open in New Window (newWindow) property is set to `false`. That is, if a link opens in the current window, all links on subsequent pages are also opened in the current window.

Filter Keyword (filterKeyword)
Identifies a keyword that items must have in order for the portlet to display a link for them. Specify the keyword. For example, if you specify `performance`, the portlet only displays links for children that have been tagged with the "performance" keyword. If the property has no value, the portlet does not filter links by a keyword.

Locale Switcher Portlet

Portlet Title	Locale Switcher
Portlet Name	wm_localeswitcher
Portlet File Name	wm_localeswitcher.pdp
Top-level Folder	development
JSR168 Portlet?	No
Alias	portlet/wm_localeswitcher
Default Instances of the portlet	None

Tools. End users use this portlet to temporarily switch their user session locale. The locale change is effective for the duration of the user session only. A page developer uses the portlet's Show List Of property to configure the locales available for selection. This list can be either all possible locales or be limited to those locales for which there are installed language packs.

Properties

Show List Of (showLangPacksOnly)
Defines the list of locales to list for selection. Specify one of the following:

- `All available locales` – Default. The portlet allows end users to switch to any available locale.

- `Installed language packs only` – The portlet allows end users to switch to only a portlet for which there is an installed language pack.

All instances of the portlet share the value of this property.

Main Nav Portlet

Portlet Title	Main Nav
Portlet Name	wm_fabric_nav___mainnav
Portlet File Name	wm_fabric_nav.war
Top-level Folder	mywebmethods
JSR168 Portlet?	Yes
Alias	wm_fabric_nav___mainnav
Default Instances of the portlet	Folders > System > Shell Sections > My webMethods Shell Header > My webMethods Main Navigation

Shared. Page developers use this portlet to display a section of the My webMethods Server taxonomy as one or more rows of tabs. It also displays the subsection of the taxonomy for each item that is contained in the first row of tabs as a hover popup-menu, that is, if the Show Menus (showMenus) property is set to true. The section displayed starts with the My webMethods Server folders configured with the Roots (roots) property and includes descendants of those roots along the path to the My webMethods Server item configured with the Leaf (leaf) property.

The levels of descendants are configured with the Depth (depth) property, for example, a depth of 3 would show the children, grandchildren, and great-grandchildren of the roots along the path to the leaf. For example, a root of Public Folders that might contain a tree view of the following items in a depth of 3:

Public Folders
  Applications
    Issue Tracker
      Create New Issue
      List Issues
    People Finder
    Tasks
      Inbox
      Outbox
  Documents
    Goals
    Reports
      Q1
      Q2
      Q3
      Q4
  Samples

Test One
Test Two

If Q1 is the leaf, and the depth is 3, then the portlet would display "Applications | Documents | Samples" children as one row of tabs, "Goals | Reports" grandchildren, that is, the grandchildren on the path to Q1 as a second row of tabs below the children, and "Q1 | Q2 | Q3 | Q4" great-grandchildren, that is, the great-grandchildren on the path to Q1 as a third row of tabs below the grandchildren. If Tasks were the leaf, then the portlet would display "Applications | Documents | Samples" children as one row of tabs, and "Issue Tracker | People Finder | Tasks" grandchildren as a second row of tabs. It would not display great-grandchildren, because the leaf was a grandchild.

In the above first example, if the Exclusive (exclusive) property is set to true, the roots would not be displayed, so the three rows of tabs would be "Applications | Documents | Samples", "Goals | Reports", and "Q1 | Q2 | Q3 | Q4". If the Exclusive (exclusive) property is set to false, the roots will be included also, so the portlet would display four rows of tabs: "Public Folders", "Applications | Documents | Samples", "Goals | Reports", and "Q1 | Q2 | Q3 | Q4".

If the Show First Level Special (showFirstLevelSpecial) property is set to true, the first level is not displayed as a row of tabs, but rather as a popup menu that lists the items in the first level. If, in using the above example, the Exclusive (exclusive) property is set to false and the Show First Level Special (showFirstLevelSpecial) property is set to true, the first row of tabs would be "Applications | Documents | Samples", the second "Goals | Reports", and the third "Q1 | Q2 | Q3 | Q4", and the left side of the first row of tabs would display a popup menu containing "Public Folders", the first level of items.

This portlet is used to display the tabs in the default "fabric" 7.x My webMethods Server shell header. The portlet instance in the header has the Show First Level Special (showFirstLevelSpecial) property set to true. It displays the roots of the My webMethods Server "fabric" taxonomy (Home, Administration, and Monitoring) in the first level "special" popup menu.

Properties

CSS Classes (css)
Provides a comma-separated list of CSS classes, one for each row. The first class applies to the first row of tabs displayed by this portlet, the second class applies to the second row, the third class to the third row, and so on. If the property has no value, it defaults to "nav,nav1,nav2".

Depth (depth)
Provides the depth of the displayed taxonomy from roots. For example, a value of "3" indicates that this portlet should show the children, grandchildren, and great-grandchildren of the roots. If the property has no value, it defaults to "3".

Exclusive (exclusive)
Indicates whether this portlet should show the roots. For example, if the roots are set to "folder.public" (the Public Folders folder), depth is set to 3, and exclusive is set to true, this portlet will show Public Folders on the first level, the children of Public Folders on the second level, the grandchildren of Public Folders on the third level, and the great-grandchildren of Public Folders on the fourth level. If the roots are set to "folder.public", depth to 3, and exclusive to false, this portlet will show the children of Public Folders on the first level, the grandchildren of Public Folders on the second level, and the great-grandchildren of Public Folders on the third level. If the property has no value, it defaults to false.

Leaf (leaf)
Provides an alias or thingID of the leaf of the displayed taxonomy. If the property has no value, it defaults to "current.resource", the alias of the currently displayed resource.

Roots (roots)
Provides a comma-separated list of aliases or thingIDs of the roots of the displayed taxonomy. If the property has no value, it defaults to "folder.public", the alias of the Public Folders folder.

Show First Level Special (showFirstLevelSpecial)
Set to true to display the first level of the taxonomy as a special dropdown menu on the left side of the first row of tabs. Set to false to display the first level as the first row of tabs. If exclusive is true, the "first level" is the children of the roots. If exclusive is false, the "first level" is the roots. If the property has no value, it defaults to false.

Show Menus (showMenus)
Set to true to show popup menus when the user hovers the pointer over the first row of tabs. Set to false to not show those menus. If the property has no value, it defaults to false.

Show Tabs (showTabs)
Set to true to show tabs for all levels of the taxonomy. Set to false to show only the first row of tabs. A user would be able to select deeper rows only by using hover popup menus. If the property has no value, it defaults to true.

Sort Key (sort)
Provides a key to sort each level. Available options are "name", "type", "modified" (that is, the last modified date), and "sortid" (that is, the sort order configured on each individual item). If the property has no value, it defaults to "name".

Sort Order (order)
Provides the order to sort each level, either "ascending" (for example, a-z, 1-10, earlier-later) or "descending" (for example, z-a, 10-1, later-earlier). If the property has no value, it defaults to "ascending".

Task Folders Only (taskFoldersOnly)
Set to true to include only task folders in the taxonomy. Set to false to include all items in the taxonomy. If the property has no value, it defaults to false.

Menubar Portlet

Portlet Title	Menubar
Portlet Name	wm_menubar
Portlet File Name	wm_menubar.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.menubar

Default Instances of the portlet	None

Page Components. Page developers add this portlet to a shell or page to provide navigation to other My webMethods Server pages.

The Menubar portlet displays the My webMethods Server taxonomy as a series of drop down menus. The page developer configures the root folder for the menu bar. For example, if the page developer configures the My webMethods Server Administration Dashboard folder as the root folder, the portlet displays the names of the six items in the Administration Dashboard folder:

Configuration | User Management | Analysis | Content | Migration | User Interface

When an end user hovers over a menu item, the portlet displays a drop down menu that contains links to the items in the corresponding folder. To navigate to an item, an end user clicks the link in the drop down menu. For example, if the user hovers over the User Management item, the portlet displays a drop down containing the following links to the portlets in the User Management folder:

- Directory Services Administration
- Locate a User's Home Folder
- Manage Groups
- Manage Roles
- Manage Users
- Principal Profile Administration

Properties

Cache Content (cacheAge)
Indicates whether to cache the content of the portlet. Specify one of the following:

- `-1` – Default. Cache the contents of the portlet. The cache isinvalidated automatically if any of the items displayed in the portlet are modified.
- `0` – Do not cache the contents of the portlet.

Visual Style (visualStyle)
Defines the layout style to use for the portlet. However, the portlet only supports one layout style. You must specify `1` for this property. The default is 1. Setting the property to any value other will break the portlet.

Orientation (orientation)
Defines how the portlet is to render the top-level items, either horizontally or vertically. Specify one of the following:

- `horizontal` – Default. Render the top-level items horizontally.
- `vertical` – Render the top-level items vertically.

Container Links (containerLinks)
Indicates whether the top-level folders are linked to the My webMethods Server folders that they represent, in addition to displaying a menu when the user hovers the pointer on the folder name. Specify one of the following:

- `true` – Default. The top-level folders link to the My webMethods Server folders that they represent, in addition to displaying a menu that lists and links to the children.
- `false` – The top-level folders are *not* linked to the My webMethods Server folders that they represent, but display a menu that lists and links to the children.

Sort Key (sort)
Identifies the key that is used to sort the items displayed, for both the top-level items and the drop-down lists. Specify one of the following:

- `name` – Default. Sort the items by the item name.
- `modified` – Sort the items by the items' last modified dates.
- `type` – Sort the items by the item type.
- *sortid* – Sort based on the sort order configured for each item. For example, the sortid can be an arbitrary integer for each item in a folder, in a nonconsecutive order, to produce a custom sort order. Items with a respective sortid of 1, 12, 15, and 10 would be sorted in the order of 1, 10, 12, and 15.

Sort Order (order)
Indicates the order in which to sort the items. Specify one of the following:

- `ascending` – Default. Sort the items in ascending order, for example, A to Z, 1 to 10, earlier to later.
- `descending` – Sort the items in descending order, for example, Z to A, 10 to 1, later to earlier.

Folder (listViewTargetURI)
Identifies the root folder to use for the menu items, for example, "folder.public". Specify the URI or alias of the root folder. By default, the portlet uses the current resource, that is, whatever page the user is currently viewing.

Links/Content/Forms Open in New Window (newWindow)
Indicates whether to open a new window when an end user clicks a top-level or drop-down menu link. Specify one of the following:

- `true` – Default. Open the link in a new window.
- `false` – Display the link in the current window.

Reuse Opened Window (reuseWindow)
Indicates whether to reuse the new window opened for the top-level or drop-down menu link. Specify one of the following:

- `true` – The portlet opens the item links associated with a top-level menu item in the same new window that it opened for the top-level menu item.

- `false` – Default. The portlet opens a new window for each item link associated with a top-level menu item.

This property is ignored if Links/Content/Forms Open in New Window (newWindow) is set to `false`. That is, if the top-level link opens in the current window, all links in its menu are also opened in the current window.

Truncate Menu Names (truncate)
Defines the maximum number of characters the portlet is to display in the names of the top-level items. If a top-level item name contains a greater number of characters than allowed, the portlet truncates it to the maximum number allowed. The portlet displays the full item names if the number of characters is equal to or less than the maximum number allowed. Specify a whole number that is `1` or higher. The default is `20`.

Mime Message Folder Renderer Portlet

Portlet Title	Mime Message Folder Renderer
Portlet Name	wm_mimemessage_renderer___mimemessagefolderrenderer
Portlet File Name	wm_mimemessage_renderer.war
Top-level Folder	extras
JSR168 Portlet?	Yes
Alias	/portlet/wm_mimemessage_renderer___mimemessagefolderrenderer
Default Instances of the portlet	None

Email. Page developers use this portlet in My webMethods Server as the default renderer for folder instances that use the data types wm_xt_nntpfolder and wm_xt_pop3folder. This portlet displays the list of e-mail messages contained in the wm_xt_nntpfolder or wm_xt_pop3folder folders.

Note:
If page developers do not want to use this default renderer, they can change the renderer for a folder instance.

Properties

Folder URI (folderURI)
Identifies the wm_xt_nntpfolder or wm_xt_pop3folder instance to display. Specify the thingID or alias of the folder instance. If the property has no value, the portlet displays an error message.

Mime Message Renderer Portlet

Portlet Title	Mime Message Renderer

Portlet Name	wm_mimemessage_renderer___mimemessagerenderer
Portlet File Name	wm_mimemessage_renderer.war
Top-level Folder	extras
JSR168 Portlet?	Yes
Alias	/portlet/wm_mimemessage_renderer___mimemessagerenderer
Default Instances of the portlet	None

Email. Page developers use this portlet in My webMethods Server as the default renderer for message instances that use the data type wm_xt_mimemessage. This portlet displays the content of an e-mail message.

Note:
If page developers to not want to use this default renderer, they can change the renderer for a message instance.

Properties

Message URI (messageURI)
Identifies the wm_xt_mimemessage instance to display. Specify the thingID or alias of the message instance. If the property has no value, the portlet displays an error message.

NNTP Folder Threaded View Portlet

Portlet Title	NNTP Folder Threaded View
Portlet Name	wm_nntpfolderthreadedview
Portlet File Name	wm_nntpfolderthreadedview.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Containers. Administrators publish an instance of this portlet to a My webMethods Server page to provide a way to view the contents of an NNTP folder (wm_xt_nntpfolder) as a list of threaded conversations. The administrator configures the portlet to reference an NNTP folder that already exists in the My webMethods Server taxonomy.

NNTP Folder Properties

Target Folder (targetFolderURI)
Identifies the NNTP folder whose contents you want to display as a list of threaded conversations. Specify the URI of the folder. If the property has no value, the portlet will contain no content.

Page Size (pageSize)
Defines the number of threads to display per page of the view. Specify a positive whole number. The default is `10`.

Pearls Skin

Component Title	Pearls Skin
Component Name	wm_skin_pearls
Component File Name	wm_skin_pearls.skin
Top-level Folder	system

Skins. This component is a skin. Page developers can use this skin as the default skin or the target of a Skin Rule. Additionally, page developers can use the skin as the base for a custom skin.

The Pearls component is a set of CSS style sheets and graphics that My webMethods Server can use when rendering a page.

The Pearls skin is used as the base for several additional sample skins, which are located in the following directory:

Software AG_directory /components/extras/ui/skin

Note:
The Pearls skin is the default skin that My webMethods Server version 7.x and later uses.

Portlet-in-Portlet Portlet

Portlet Title	Portlet-in-Portlet
Portlet Name	wm_portletinportlet
Portlet File Name	wm_portletinportlet.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.portletinportlet

Default Instances of the portlet	None

Page Components. A page developer adds this portlet to a page to display the contents of another page within the portlet.

For example, if the page developer adds this portlet to the Public Folders page and configures it to display the current user's home folder, when an end user views the Public Folders page it would contain that user's home folder.

Properties

Portal Page (proxyThingID)
Identifies the page to display. Specify the URI or alias of the page. If the property has no value, the portlet displays no content.

Renderer (subrenderer)
Indicates how to display the portlets within the page, either altogether like a shell section or separately, each with their own title bar and portlet borders like a portal page. Specify one of the following:

- `portletinportlet` – Default. Display all portlets in the page together like a shell section.
- `pageinportlet` – Display the portlets in the page each with their own title bar and portlet border like a portal page.

Auto-Positioned Portlets (displayHobos)
Indicates whether to display or hide auto-positioned portlets, which are portlets that are not specifically positioned in a row and column. Specify one of the following:

- `true` – Display auto-positioned portlets.
- `false` – Default. Hide auto-positioned portlets.

Only Display Portlets With These Attributes (displayOnly)
Identifies the portlets in the page that should be rendered, based on the custom attributes of the rows or columns containing those portlets. Each portlet on the page will be rendered only if that portlet is contained by a row or column that has at least one of the attribute values specified by this property.

Specify the attributes in a comma-separated values (CSV) format. If the property has no value, all portlets in the page will be rendered, except hobos if the displayHobos property is set to true, or portlets in rows or columns with the attributes specified by the displayNot property.

Don't Display Portlets With These Attributes (displayNot)
Identifies the portlets in a page that should *not* be rendered, based on the custom attributes of the rows or columns containing those portlets. Each portlet on the page will be rendered only if it is contained by a row or column that does not have any of the attribute values specified by this property.

Specify the attributes in a comma-separated values (CSF) format. If the property has no value, all portlets in the page will be rendered, except hobos if the displayHobos property is set to true, or portlets in rows or columns without the attributes specified by the displayOnly property.

Preformatted Text Portlet

Portlet Title	Preformatted Text
Portlet Name	wm_preformattedtext
Portlet File Name	wm_preformattedtext.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.preformattedtext
Default Instances of the portlet	None

Drawing. Page developers use this portlet to display text content using plain-text formatting.

Properties

Text (text)
Holds the text content to display in the portlet. If the property has no value, the portlet displays nothing.

Width (width)
Defines a fixed width for the panel that displays the formatted text. Specify a valid CSS width measurement, for example, `350px`, `30em`, or `50%`. If the formatted text requires more horizontal space than the panel allows, the portlet either wraps long text or displays a horizontal scroll bar; for more information, see the Word Wrap (wordwrap) property. If the property has no value, the panel uses 100 percent of the available width.

Height (height)
Defines a fixed height for the panel that displays the formatted text. Specify a valid CSS height measurement, for example, `350px` or `30em`. If the formatted text requires more vertical space than the panel allows, the portlet displays a vertical scroll bar. If the property has no value, the panel grows vertically to show the entire contents without a vertical scroll bar.

Word Wrap (wordwrap)
Indicates whether to wrap long lines or to display a horizontal scroll bar. Specify one of the following:

- `true` – Default. Wrap long lines to the next line if there is not enough horizontal space.
- `false` – Display a horizontal scroll bar if there is not enough horizontal space.

Monospace (monospace)

Indicates whether to use a monospace font to display the formatted text. Specify one of the following:

- `true` – Use a monospace font.
- `false` – Default. Do not use a monospace font.

Cache Content (cacheAge)
Indicates whether to cache the formatted text. Caching the formatted text speeds up subsequent renderings of the text. Specify one of the following:

- `-1` – Cache the formatted text.
- `0` – Default. Do not cache the formatted text.

Publish Portlet

Portlet Title	Publish
Portlet Name	wm_publish
Portlet File Name	wm_publish.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.publish
	/portlet/wm_publish
Default Instances of the portlet	Administrative Folders > Administration Dashboard > Portal Content > Publish

Page Authoring. Page developers and end users use this portlet to publish new instances of My webMethods Server objects. The base types of the My webMethods Server objects that can be published are files, folders, form, links, and portlets.

The following table lists the base types along with the custom subtypes each can be extended by:

Base types of objects to publish	Subtypes the base type can be extended by
File	Dynamic Business Objects (DBOs)
Folder	Dynamic Business Objects (DBOs)
Form	Dynamic Business Objects (DBOs)
Link	Dynamic Business Objects (DBOs)
Portlets	Portlets

Note:

Page developers and end users can also publish My webMethods Server objects using the Page Editor for the target folder.

Note:
To simplify the publish process for end users, page developers can use the PublishContext Portlet that allows the page developer to pre-set fields in the Publish Portlet wizard.

PublishContext Portlet

Portlet Title	PublishContext
Portlet Name	wm_publishcontext
Portlet File Name	wm_publishcontext.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	/portlet/wm_publishcontext
Default Instances of the portlet	None

Content Management. Page developers use this portlet to simplify the publish process for end users. The PublishContext portlet is a helper for the Publish Portlet, which is a wizard for adding new content to the My webMethods Server content repository. Use this PublishContext portlet to render a button, link, or image control on a page so that when a user clicks the control, the application opens a page that contains the Simple Publish Portlet with the fields of the wizard pre-set. The PublishContext portlet allows you to pre-populate field values, make fields read-only, and/or hide fields in the Simple Publish Portlet wizard.

Properties in the Portlet Preferences

Cache Age (cacheAge)
Indicates how long, in minutes, to keep the rendered portlet in cache. All instances of the PublishContext portlet share the value of this property.

The following lists the values that are available in the user interface for this property:

- `0` – Default. Do not cache potlet.
- `1` – Cache for 1 minute.
- `10` – Cache for 10 minutes.
- `15` – Cache for 15 minutes.
- `20` – Cache for 20 minutes.
- `30` – Cache for 30 minutes.

- `60` – Cache for 1 hour.
- `-1` – Cache infinitely.

Link User Interface Properties

Link Type (publishLinkType)
Indicates whether you want to use a button, link, or image control. Specify one of the following:

- `button`
- `link`
- `image`

The default is `button`.

Link Text (publishLinkText)
Defines the label to use for a button or link control. Use this property when **Link Type** is set to `button` or `link`. Specify any text. The default value is:

```
Publish new ...
```

Link Image (publishLinkImage)
Identifies the image file to use for an image control. Use this property when **Link Type** is `image`.

Specify a URL for the image or relative path to an addressable image in My webMethods Server. If **Link Type** is `image` and you do not specify a value for **Link Image**, or if the value for **Link Image** is not valid, a broken image icon is displayed.

Publishing Context Properties

DBO/Portlet Type (xTypeThingID)
Restricts the types of resources (files, folders, links, or pages) that an end user can publish using the Simple Publish Portlet wizard. For example, if you want to allow end users to only publish link objects, set this property to `link`.

You must specify a thingID or an alias of a DBO (xtype); otherwise, the portlet type for this property `file`, `folder`, `link`, or `page` will not work. You can use the following aliases, or any other alias to DBO or portlet types:

- `/xtype/link` – Alias to the base link DBO.
- `/xtype/content` – Alias to the base file DBO.
- `/xtype/folder` – Alias to the base folder DBO.

Note:
You can specify only one thingID or alias.

Publish DBO Properties (publishProperties)
Pre-sets the default values for the properties displayed in the Simple Publish Portlet wizard. Each DBO or Portlet Type has its own set of properties.

The format for this property is a semicolon-separated list of property settings, and each property setting itself consists of a comma-separated list of four items:

Property	Description
`name`	The internal name of the property, for example, "description" is the name property setting for the description property.
`value`	The value for the property, for example, "published via a custom Public Context portlet".
`hidden`	Indicates whether the property should be hidden or not. The hidden property setting can be either true, false, or default. **Note:** The default setting indicates that this property should be hidden if so configured in the DBO or portlet's default configuration.
`readonly`	Indicates whether the property should be read-only or not. The readonly setting can be either true, false, or default. **Note:** The default setting indicates that this property should be read-only if so configured in the DBO or portlet's default configuration.

Note:
There is no default property setting. If the property has no value, DBO or Portlet Type properties will not have default values.

The following example of publishProperties specifies three properties:

Example Property	Description
`parentID` (Parent Folder)	Specifies that this property should be set to "folder.public" (Public Folders), and that the property should not be displayed to or be editable by the user.
`name`	Specifies that the "name" property should default to "New Folder", but that the user can edit this property.
`description`	Specifies that the "description" property should be "created via a custom Publish Context portlet", and that while the property and its value will be displayed to the user (hidden is false), the user will not be able to edit it (readonly is true).

```
parentID,folder.public,true,true;name,New
Folder,false,false;description,created via a custom Publish Context
portlet,false,true
```

Publish Finish URL (publishFinishURL)

Identifies the page to display after the Simple Publish Portlet wizard completes normally. Specify the URL of the page. There is no default. If no publishFinishURL is specified, the application displays the page that the user last viewed last before starting the Simple Publish Portlet wizard.

Publish Cancel URL (publishCancelURL)
Identifies the page to display when the end user clicks the Cancel button in the Simple Publish Portlet wizard. Specify the URL of the page. There is no default. If no publishCancelURL is specified, the application displays the page that the user viewed last before starting the Simple Publish Portlet wizard.

Quick Access URL Portlet

Portlet Title	Quick Access URL
Portlet Name	wm_quickaccessurl
Portlet File Name	wm_quickaccessurl.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Content Management. Page developers use this portlet to display an intuitive URL for the page. This intuitive URL is built from the shortest alias to the page. If the page has no alias, this portlet displays nothing.

Properties

Display Style (Style)
Defines the style to use to display the URL. Specify one of the following:

- `Label (style.default)` – Default. Prints "Quick Access URL: http://localhost:8585/folder.public".
- `Parenthesis (style.parenthesis)` – Prints "(http://localhost:8585/folder.public)".
- `Brackets (style.brackets)` – Prints "[http://localhost:8585/folder.public]".
- `Angle-Brackets (style.brackets)` – Prints "<http://localhost:8585/folder.public>".

Renderer Tabs Portlet

Portlet Title	Renderer Tabs
Portlet Name	wm_rendertabs

Portlet File Name	wm_rendertabs.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	/portlet/wm_rendertabs
	Note: Do not use this instance. Instead publish another instance into the My webMethods Server to use.
Default Instances of the portlet	None

Content Management. End users use this portlet to instantly switch the renderer for the current resource (object) being browsed to another renderer. The intended use of the portlet is to include it as part of the shell.

A renderer is typically a portlet. Its purpose is to implement a user interface for a specific My webMethods Server object. My webMethods Server includes many custom renderers. Renderers can be:

- **Generic renderers that apply to any kind of server object.** For example, the Permissions renderer used for the permissions screen. All objects have permissions, so the Permissions renderer can be applied to all objects. Another generic renderer is the Properties renderer, which is used for the properties screen.
- **Specialized renderers that apply to only specific types of objects.** Additionally, new renderers can be created as part of custom applications.

An administrator places the Renderer Tabs portlet in a shell and configures its properties to define the visual style for selecting a renderer, the renderers to allow for selection, and for what objects to display the Renderer Tabs portlet.

When the Renderers Tabs portlet is available for a resource, the end user can instantly switch the renderer type for the current resource. For example, select the Permissions renderer to view the permissions screen for an object, or select the Details renderer to display the details of the current resource as a simple list rather than a web page.

Properties

Style (style)
The available styles depend on the deployed tab styles, for example, wm_tabstyles_popup, wm_tabstyles_spotlight, and so on. The following table lists the default styles.

Style	Description

`Tabs`
 	
`(portlet.tabstyles)`
	Default. Displays standard tabs as described in Spotlight Tab Styles Portlet.

Style	Description
`Subtabs` `(portlet.tabstyles?style=condensed)`	Displays condensed tabs.
`Horizontal` `(portlet.tabstyles?layout=horizontal)`	Displays a horizontal list of links.
`Vertical` `(portlet.tabstyles?layout=vertical)`	Displays a vertical list of links.
`Horizontal - Bars` `(portlet.tabstyles?layout=horizontal&separator=|)`	Displays a horizontal list of links, with each link separated by a vertical bar.

`Popup`
`(portlet.tabstyles.popup?text=[selected]&`
`defaultText=[name])`
	Displays a single link that shows the selected renderer. When clicked, a pop-up menu displays to the side of the link, showing all available renderers, as described in Popup Tab Styles Portlet.

`Dropdown`
`(portlet.tabstyles.popup?text=[name]&`
`dropdown=true)`
	Displays a single link that shows the name of the portlet instance. When clicked, a pop-up menu displays below the link, showing all available renderers.

`Hover`
`(portlet.tabstyles.popup?text=[name]&`
`layout=hover)`
	Displays a single link that shows the name of the portlet instance. When hovered over, a pop-up menu displays below the link, showing all available renderers.

Renderer Properties

Available Renderers (renderers)

Defines the list of renderers that you want to make available for rendering a resource. End users select one of these renderers to instantly change how the current resource is rendered. Specify the list of renderers, by using a comma-separated list of renderer names. For example, you might specify `permissions, properties, details` to make the available renderers the Permissions, Properties, and Details renderers. If the property has no value, only the default renderer for the current resource will be available.

Types to Use (Xtypes)

Defines the types of My webMethods Server objects for which to display the Renderer Tabs portlet. Specify the list of objects, by using a comma-separated list of type names. For example, you might specify folder so that the system only displays the Renderer Tabs portlet for folders. If the property has no value, the system will display the Renderer Tabs portlet for all object types.

Roots (roots)

Controls where the system displays the Renderer Tabs portlet. Specify a comma-separated list of thingIDs or aliases of the root folders in the My webMethods Server taxonomy where the Renderer Tabs portlet should be displayed. The system displays the Renderer Tabs portlet on the shell for all objects under the root folders you specify. For example, if you set **Roots** to `Public Folders` and the current object is under Public folders, the Renderer Tabs portlet will be visible. If the property has no value, the system displays the Renderer Tabs portlet everywhere in the My webMethods Server taxonomy provided that the current shell includes the Renderer Tabs portlet.

Session History Portlet

Portlet Title	Session History
Portlet Name	wm_sessionhistory.
Portlet File Name	wm_sessionhistory.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Page Components. Page developers add this portlet to a portal shell to provide end users with links to pages they have previously visited during their login session. End users click links to navigate back to pages in their history.

History Info Properties

History Length (historyLength)
Defines the number of links to keep in the history queue. The portlet adds new links to the end of the queue. When the number of links specified by **History Length** has been reached and a new link is added, the portlet removes one link from the beginning of the queue to maintain the specified number of links in the queue. Specify `5`, `10`, `15`, or `20`. The default is `5`.

Truncate at n characters (charTruncate)
Defines the maximum number of characters you want the portlet to display for the name of each link in the history. Specify a positive whole number. There is no upper limit. The default is `15`.

Style to display the history in (displayStyle)
Indicates the style to use to display the links in the portlet. Specify one of the following:

- `Horizontal` – Default. The portlet displays the links horizontally.
- `Vertical` – The portlet displays the links vertically.
- `Drop Down` – The portlet displays the links in a drop-down list.

Link Separation Character (linkCharSeperator)

Indicates the character to use to separate the links displayed in the portlet. This property is only used when the Style to display the history in (displayStyle) property is Horizontal. The default is a space character.

Simple Publish Portlet

Portlet Title	Simple Publish
Portlet Name	wm_simplepublish
Portlet File Name	wm_simplepublish.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.publish.simple
	/portlet/wm_publishsimple
Default Instances of the portlet	None

Page Authoring. Page developers and end users use this portlet to publish new instances of links, documents, folders, and pages.

Note: My webMethods Server uses this portlet internally for the New context menu.

Properties

Type (type)
Required. Identifies the type of My webMethods Server object to publish. Specify one of the following:

- folder
- link
- content

Container (parentID)
Identifies the folder into which you want to publish the My webMethods Server object. Specify the URI or alias of the folder. The default is folder, which indicates that the portlet is to publish the My webMethods Server object into the current My webMethods Server folder.

Container (parentID)
Identifies the folder into which you want to publish the My webMethods Server object. Specify the URI or alias of the folder. The default is `current.resource`, which indicates that the portlet is to publish the My webMethods Server object into the current My webMethods Server folder.

Simple View Portlet

Portlet Title	Simple View
Portlet Name	wm_simplelink
Portlet File Name	wm_simplelink.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.simplelink
Default Instances of the portlet	None

Content Management. Page developers use this portlet to display the name of and/or link to another item in the My webMethods Server content repository.

Properties

Visual Style (visualStyle)
Indicates whether you want the portlet to display the item name with or without a link. Specify one of the following:

- `1` – Default. The portlet displays the item name with a link.
- `2` – The portlet displays the item name without a link.

Portal Item (proxyThingID)
Identifies the item in the My webMethods Server content repository for which to render the simple view. Specify the thingID or alias of the item. If the property has no value, the portlet displays nothing.

Truncate Name After X Characters (truncate)
Defines the number of characters to display for an item name. If the item name contains more characters than the value you specify, the portlet truncates the displayed name. Specify `10`, `15`, `20`, `30`, `40`, `50`, or `100`. The default is `50`.

Open in New Window (newWindow)
Indicates whether you want the portlet to open a new window when an end user clicks an item link. Specify one of the following:

- `true` – Default. Open a new window for the item link.
- `false` – Display the item link in the current window.

Reuse Opened Window (reuseWindow)
Indicates whether to reuse the opened window for an item link if the end user clicks the link again. Specify one of the following:

- `true` – Re-use the open window.
- `false` – Default. Open another new window for the item link.

This property is ignored if Open in New Window (newWindow) is set to `false`.

HTML ID (htmlId)
Defines an ID for the HTML element for the item. Specify a value you want to assign to the HTML ID attribute. This is useful if you need to reference the item link from a client-side script or enable custom CSS styles. If the property has no value, the portlet applies no ID to the HTML element for the item.

HTML Class (htmlClass)
Identifies the CSS class to apply to the HTML element for the item. Specify any valid CSS class name. If the property has no value, the portlet applies no CSS class to the HTML element.

HTML Style (htmlStyle)
Identifies the CSS style to apply to the HTML element for the item. Specify any valid CSS style value. If the property has no value, the portlet applies no CSS style to the HTML element.

Streaming Content Viewer Portlet

Portlet Title	Streaming Content Viewer
Portlet Name	wm_streamingcontentviewer
Portlet File Name	wm_streamingcontentviewer.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_streamingcontentview
Default Instances of the portlet	None

System. Page developers publish this portlet to a My webMethods Server page to display a flash widget or streaming media. After publishing the portlet to a page, the page developer configures it to point to a media file that is published elsewhere in My webMethods Server. Based on the file type of the media file, the Streaming Content Viewer portlet uses either the Flash player or Windows Media player to display the media.

Resource Properties

Resource (contentURI)
Identifies the media file to display. Specify the thingID or alias to the media file. When configuring the portlet from the user interface, use the resource picker to select the file to display. The media file must be published into a My webMethods Server folder. If the property has no value, the embedded viewer will display nothing.

Surrogate Portlet

Portlet Title	Surrogate Portlet
Portlet Name	wm_surrogateportlet
Portlet File Name	wm_surrogateportlet.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	/portlet/wm_surrogateportlet
Default Instances of the portlet	None

Content Management. Page developers use this portlet to render the user interface of a portlet that resides in another My webMethods Server page. This portlet provides a simple way to share a portlet instance among two or more pages.

Properties

Proxy Thing (proxyThingID)
Identifies the portlet instance that you want to render within the Surrogate Portlet. Specify the thingID or alias of the portlet. If the property has no value, the Surrogate Portlet displays the warning text "Proxy has not been configured".

Tabs View Portlet

Portlet Title	Tabs View
Portlet Name	wm_tabsview
Portlet File Name	wm_tabsview.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.tabsview
Default Instances of the portlet	None

Page Components. Page developers use this portlet to display the contents of a My webMethods Server folder. The portlet displays each item within the folder as a tab. Additionally, the portlet displays the contents of the item associated with the selected tab.

For example, if the Tabs View portlet is configured to display "folder.admin.dashboard.portal_analysis" (i.e., the Analysis folder of the Administration Dashboard), the Tabs View portlet would display three tabs, one for each item in the Analysis folder: Logging Configuration, Logging Viewer, and Session Monitor. When a tab is selected, for example, the Logging Configuration tab, the portlet would display the Logging Configuration portlet.

Properties

Folder (proxyThingID)
Identifies the folder whose items you want displayed in the Tab View portlet. Specify the thingID or alias of the folder. If this property has no value, the Tab View portlet displays nothing.

Style (style)
Indicates the tab style to use. This portlet uses the Tabs (wm_tabs) portlet to render the tabs. The Tabs portlet allows arbitrary tab styles, such as those provided with the wm_tabstyles portlet, the Popup Tab Styles Portlet, and the Spotlight Tab Styles Portlet. The following lists examples of tab styles you can specify:

- portlet.tabstyles

  (Tabs) The portlet uses the default My webMethods Server-style tabs. This is the default.

- portlet.tabstyles?style=titlebar-tabs

  (Titlebar Tabs) The portlet uses the default My webMethods Server-style tabs with a title bar background.

- portlet.tabstyles?style=condensed

  (Subtabs) The portlet displays second-level tabs.

- portlet.tabstyles?style=horizontal

  (Horizontal) The portlet displays the tabs as links placed horizontally with a space character used to separate the links.

- portlet.tabstyles?style=vertical

  (Vertical) The portlet displays the tabs as links placed vertically.

- portlet.tabstyles?style=horizontal&separator=*char*

  (Horizontal - Bars) The portlet displays the tabs as links placed horizontally with a character you specify to separate the links. Specify any character for *char*, for example, | for the pipe character, / for a slash, or : for a colon.

- portlet.tabstyles.popup?text=[selected]&defaultText=[name]

  (Popup) The portlet displays the tabs as a popup menu that is activated when a user clicks on the target text. The popup menu displays to the left of the target text. When there is a selected tab, the portlet displays the title of the selected tab.

- portlet.tabstyles.popup?text=[name]&dropdown=true

(Dropdown) The portlet displays the tabs as a popup menu that is activated when a user clicks on the target text. The popup menu displays as a dropdown list below the target text. When there is a selected tab, the portlet displays the instance name of the Tabs (wm_tabs) portlet.

- portlet.tabstyles.popup?text=[name]&layout=hover

  (Hover) The portlet displays the tabs as a popup menu that is activated when a user hovers on the target text. The portlet displays the tabs as a popup menu that displays to the left of the target text. When there is a selected tab, the portlet displays the instance name of the Tabs (wm_tabs) portlet.

- portlet.tabstyles.spotlight?layout=horizontal

  (Horizontal - Spotlight) The portlet displays the tabs as a horizontal list of links with the selected tab link displayed in the skin's "spotlight" style.

- portlet.tabstyles.spotlight?layout=vertical

  (Vertical - Spotlight) The portlet displays the tabs as a vertical list of links with the selected tab link displayed in the skin's "spotlight" style.

- portlet.tabstyles.spotlight?layout=horizontal&selectAll=true

  (Horizontal - Spotlight All) The portlet displays the tabs as a horizontal list of links with all tab links displayed in the skin's "spotlight" style.

- portlet.tabstyles.spotlight?layout=vertical&selectAll=true

  (Vertical - Spotlight All) The portlet displays the tabs as a vertical list of links with the all tab links displayed in the skin's "spotlight" style.

Tabs Per Page (pageSize)
Defines the maximum number of folder items to display as tabs. Specify a positive whole number. The default is `10`. To display all folder items, specify a number that exceeds the total number of folder items, such as 1000.

Sort Key (sort)
Identifies the key to use to sort the tabs. Specify one of the following:

- `name` – Default. Sort the tabs based on the item names.
- `type` – Sort the tabs based on the item types.
- `modified` – Sort the tabs based on the items' last modified dates.
- `sortid` – Sort based on the sort order configured for each item. For example, the sortid can be an arbitrary integer for each item in a folder, in a nonconsecutive order, to produce a custom sort order. Items with a respective sortid of 1, 12, 15, and 10 would be sorted in the order of 1, 10, 12, and 15.

Sort Order (order)
Indicates the order in which to sort the tabs. Specify one of the following:

- `ascending` – Default. Sort the tabs in ascending order (i.e., A to Z, 1 to 10, earlier to later).

- `descending` – Sort the tabs in descending order (Z to A, 10 to 1, later to earlier)

Thing Navigate Up or Down Button Portlet

Portlet Title	Thing Navigate Up or Down Button
Portlet Name	wm_thingmaximize
Portlet File Name	wm_thingmaximize.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_thingmaximize
	portlet.thingmaximize
Default Instances of the portlet	None

Title Bar Tools. Page developers use this portlet as part of a shell title bar to provide a way to navigate to the My webMethods Server resource parent folder or back to the current resource.

Note:
This portlet is used internally in several shell title bars.

Properties

Target (target)
Identifies the My webMethods Server object to which to navigate. Specify the URI of the object. If the property has no value, the default value `current.resource` is used, which means the target is the currently opened My webMethods Server resource. Instances of the portlet do not share the value of this property.

Action (maximize)
Indicates the action to take. Specify one of the following:

- `true` – Default. The action is up to navigate to the parent resource.
- `false` – The action is down, to navigate to "self"; that is, the current resource.

Instances of the portlet do not share the value of this property.

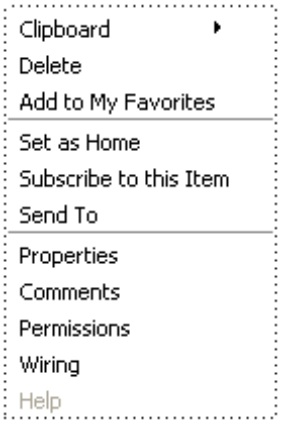
Thing Popup Menu Portlet

Portlet Title	Thing Popup Menu
Portlet Name	wm_thingpopup

Portlet File Name	wm_thingpopup.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_thingpopup
Default Instances of the portlet	None

Title Bar Tools. Page developers can use this portlet in custom title bars to provide a default popup menu for a My webMethods Server object. This portlet is part of the core system components and end users should not directly access it.

This portlet adds the  to the title bar. An end user can click this icon to access the following popup menu:



Properties

Target (target)
Required. Identifies the target My webMethods Server object for which the popup menu should be displayed. Specify the ID or alias of the target object.

Cache Content (cacheAge)
Indicates whether the HTML contents of popup menu should be cached. Specify one of the following:

- `-1` – Default. Cache the content infinitely.
- `0` – Do not cache the content.
- `1` – Cache the content for 1 minute.

Thing Portlet Mode Buttons Portlet

Portlet Title	Thing Portlet Mode Buttons
Portlet Name	wm_titlebar_tools_portletmode
Portlet File Name	wm_titlebar_tools_portletmode.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_titlebar_tools_portletmode
	portlet.titlebar_tools_portletmode
Default Instances of the portlet	None

Title Bar Tools. Page developers can use this portlet in custom title bars to add icons that allow end users to switch the portlet mode. The supported portlet modes are: view (default), edit, help.

This portlet is part of the core system components; internally My webMethods Server portlet title bars and shell title bars use this portlet.

Properties

Target (target)
Identifies the portlet to which to add the portlet mode buttons. Specify the URI of the portlet. The portlet can be a legacy portlet or a JSR168 portlet. If the property has no value, the default value `current.resource` is used, which means the portlet mode buttons are added to the currently opened My webMethods Server resource and portlet. Instances of the portlet do not share the value of this property.

Thing Window State Buttons Portlet

Portlet Title	Thing Window State Buttons
Portlet Name	wm_titlebar_tools_windowstate
Portlet File Name	wm_titlebar_tools_windowstate.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_titlebar_tools_windowstate
	portlet.titlebar_tools_windowstate

Default Instances of the portlet	None

Title Bar Tools. Page developers can use this portlet in custom title bars to add icons that allow end users to switch portlet window states. The supported portlet window states are: normal (default), minimized, maximized.

This portlet is part of the core system components; internally My webMethods Server portlet title bars and shell title bars use this portlet.

Properties

Target (target)
Identifies the portlet to which to add the portlet window state buttons. Specify the URI of the portlet. The portlet can be a legacy portlet or a JSR168 portlet. If the property has no value, the default value `current.resource` is used, which means the portlet window state buttons are added to the currently opened My webMethods Server resource and portlet. Instances of the portlet do not share the value of this property.

Time Zone Portlet

Portlet Title	Time Zone
Portlet Name	wm_timezone
Portlet File Name	wm_timezone.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.timezone
Default Instances of the portlet	None

Page Components. End users use this portlet to determine the time zone of the displayed page.

Properties

Display Style (mode)
Indicates whether to add a prefix label of "All Times". For example, the portlet displays "All Times PST (UTC-8:00)" instead of just "PST (UTC-8:00)". Specify one of the following:

- `Basic` – Default. The prefix label "All Times" is not added.
- `Explanatory` – Adds the prefix label "All Times".

Titlebar Guest Titlebar Portlet

Portlet Title	Titlebar for the Guest Shell
Portlet Name	wm_titlebar_guest
Portlet File Name	wm_titlebar_guest.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_titlebar_guest
Default Instances of the portlet	None

Title Bar Tools. Page developers can use this portlet when implementing a custom shell to provide a simple title bar that displays only the portlet title without any actions or menus.

This portlet is part of the core system components. It is used for the Guest Shell, which is the default shell for all non-authenticated users. End users should not directly access or publish this portlet. Page developers can us it when developing a custom Guest Shell or any other shell that requires a title bar without tools.

Properties

Cache Age (cacheAge)
Required. Indicates how long to cache the contents of the portlet. Specify one of the following:

- `-1` – Default. Cache the contents of the portlet infinitely.
- `0` – Do not cache the contents of the portlet.
- `1` – Cache the contents of the portlet for 1 minute.
- `10` – Cache the contents of the portlet for 10 minutes.
- `15` – Cache the contents of the portlet for 15 minutes.
- `20` – Cache the contents of the portlet for 20 minutes.
- `30` – Cache the contents of the portlet for 30 minutes.
- `60` – Cache the contents of the portlet for 60 minutes.

All instances of the portlet share the value of this property.

4 Dynamic Business Objects

This chapter describes the dynamic business objects (DBO)s that My webMethods Server provides out-of-the-box.

Summary of DBOs

The following table lists the dynamic business objects (DBO)s that My webMethods Server provides out-of-the-box.

Name	Description
AuthScheme DBO	Dynamic Business Objects. Portlet developers use objects of this type to define auth-handlers.
Ftp Folder DBO	Containers. End users use this type of folder to perform a bulk update or upload of files to a My webMethods Server folder from a remote FTP client.
Http Header Policy DBO	Security. Administrators use this security provider to configure permissions on My webMethods Server items. Unlike other security providers where privileges are manually assigned, use this security provider to dynamically grant or deny access to server objects based on information that is passed to My webMethods Server in the HTTP header when a user logs in.
Locale Property Storage DBO	Dynamic Business Objects. The system uses this system dynamic business object (DBO) to implement multi-locale property value storage. When implemented for a portlet, an end user can define different property values based on locales. My webMethods Server automatically publishes instances of this Locale Property Storage DBO; users cannot publish it.
Mime Message DBO	Dynamic Business Objects. The system uses an instance of this data type in My webMethods Server to publish an e-mail message that was retrieved from an e-mail inbox or NNTP newsgroup on an external server to a file in a folder. My webMethods Server typically uses this data type when synchronizing an e-mail inbox or NNTP newsgroup to a wm_xt_nntpfolder and wm_xt_pop3folder folder instance.
My webMethods Folder DBO	Dynamic Business Objects. Page developers use this dynamic business object to provide additional data to folders such as whether it is an "isTaskFolder".
NNTP Folder DBO	Containers. Administrators publish an instance of this folder type to a My webMethods Server to provide a way to view the contents of an NNTP newsgroup. The NNTP Folder polls an NNTP newsgroup and creates entries within the folder for each newsgroup message it retrieves.

Name	Description
Pop3 Folder DBO	Containers. Administrators publish an instance of this folder type to a My webMethods Server page to provide a way to view the contents of a POP3 mailbox. The Pop3 Folder polls a POP3 mailbox and creates entries within the folder for each e-mail message it retrieves.
Portal Skin DBO	Dynamic Business Objects. Page developers use this object type to define a new skin and configure its properties.
Relative Link DBO	Dynamic Business Objects. Page developers use instances of this type to add links that are relative to the front-end server to a page. To provide a set of links for the end user, page developers publish multiple instances of this link type to a folder.
Restful Link DBO	Dynamic Business Objects. Page developers use this portlet to build a link to another My webMethods Server page. In addition to linking to another page, this portlet also allows the page developer to supply initial values of properties for a portlet on the target page.
Role DBO	Dynamic Business Objects. Administrators use this dynamic business object (DBO) in My webMethods Server to store general role properties for a role. My webMethods Server implicitly creates instances of this DBO when new roles are added to the system.
Shell DBO	Dynamic Business Objects. Page developers use this object type to define a new shell and configure its header, footer, left navigation, and right navigation.
Shortcut DBO	Dynamic Business Objects. Page developers use instances of this type to define shortcuts to My webMethods Server resources.
Static Role Provider DBO	Dynamic Business Objects. Administrators use this dynamic business object (DBO) in My webMethods Server to store configuration for static roles. My webMethods Server implicitly creates instances of this DBO when new static roles are added to the system.
Website Archive Viewer DBO	Page components. Page developers use this File dynamic business object (DBO) to upload a static web site archive (either a .zip or .jar), and to publish the web site archive so that end users can browse its contents within the My webMethods Server taxonomy.

AuthScheme DBO

Portlet Title AuthScheme

Portlet Name	wm_xt_authscheme
Portlet File Name	wm_xt_authscheme.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Dynamic Business Objects. Portlet developers use objects of this type to define auth-handlers.

Properties

Visual Style (visualStyle)
Identifies the JSP file to use to render this auth-scheme's properties. The default value is `dbo_view.jsp`. All instances of the portlet share the value of this property.

General Properties

Name (name)
Required. Defines the name for this auth-scheme. Specify a string that contains less than 100 characters.

Description (description)
Defines a description for this auth-scheme. Specify a string that contains less than 255 characters.

Keywords (keywords)
Defines the keywords for this auth-scheme. Users can later use these keywords to search for this auth-scheme by using My webMethods Server search capabilities. The length of the entire list of keywords must be less than 255 characters.

Location Properties

Parent Folder (parentID)
Required. Identifies the folder where this auth-scheme is stored. Specify the thingID or alias of the folder, for example, folder.auth.schemes.

Extended Properties

Authentication Scheme ID (authSchemeLevel)
Required. Identifies this auth-scheme. This ID must be a unique integer among the auth-schemes deployed to My webMethods Server. Auth-schemes built into My webMethods Server use numbers less than 10.

AuthScheme Runtime ClassName (challengeResponseClassName)
Required. Defines the class of the auth-handler. Specify the class, including path.

Performs Redirect (hasRedirect)
Indicates whether to use the Redirect URI (redirectURI) property. Specify one of the following:

- `true` – Use the Redirect URI (redirectURI) property.
- `false` – Do not use the Redirect URI (redirectURI) property.

Redirect URI (redirectURI)
Identifies the page that provides the auth-handler change input. For example, for forms-auth the redirectURI would be a login page. Specify the URI of the page. If the property has no value, or hasRedirect is false, the auth-handler challenge will return the requested page, instead of redirecting.

Ftp Folder DBO

Portlet Title	Ftp Folder
Portlet Name	wm_xt_ftpfolder
Portlet File Name	wm_xt_ftpfolder.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	/portlet/wm_xt_ftpfolder
	Note: Do not use this instance. Instead publish another instance into My webMethods Server.
Default Instances of the portlet	None

Containers. End users use this type of folder to perform a bulk update or upload of files to a My webMethods Server folder from a remote FTP client.

The Ftp Folder behaves like a typical My webMethods Server folder or page, but it starts an FTP server that uses a configured FTP port. From a remote FTP client, an administrator uses the folder to manipulate (for example, create, delete, or update) My webMethods Server folders, pages, and content files as if they are file system folders and files. The administrator cannot use the Ftp Folder to manipulate My webMethods Server portlets.

To use the Ftp Folder, administrators must log into the FTP server providing valid credentials for a My webMethods Server user account. The folder does not support anonymous access. Additionally, the My webMethods Server user account must have at least READ access to the folders, pages, and content files being manipulated.

You can publish instances of the Ftp Folder in the following ways:

- **When opening a folder or page for editing**. From a My webMethods Server page:

1. Use the **Tools** popup menu and select the **Edit Page** command.
2. Click the **Containers** link in the editor palette.
3. Drag the Ftp Folder on to the page.
4. Specify properties for the folder. For more information, see General Properties and Extended Properties below.

- **When publishing a folder.**

1. Access the Publish portlet: **Administration > Content > Publish**.
2. Select **Folder** for the type of object to publish and then select the subtype **Ftp Folder**.
3. Select the parent folder or page where the Ftp Folder is to be placed.
4. Specify properties for the folder. For more information, see General Properties and Extended Properties below.

General Properties

Name (name)
Required. The name of the folder, as it will be displayed in the My webMethods Server taxonomy. Specify the name of an object in the taxonomy. You can specify from 1 through 100 characters.

Description (description)
Provides a description of the folder in the My webMethods Server taxonomy. Must be less than 255 characters long. If the property has no value, no description will be displayed for the folder.

Keywords (keywords)
Provides a list of keywords to assign to the folder in the My webMethods Server taxonomy. Users can later use these keywords to search for this folder in the taxonomy. Use 0 through 255 characters for keywords. If the property has no value, no keywords will be displayed for the folder.

Extended Properties

Ftp Port (port)
Required. Identifies the FTP port to use when starting the FTP server for the Ftp Folder portlet. Specify a port number that is globally unique among all Ftp Folder portlets published in My webMethods Server. Also ensure the port that you specify does not conflict with any other ports on the physical server running My webMethods Server. The default FTP port is 8021.

Folder Type (specialFolderType)
This folder can make a different folder available by means of FTP. By default, the folder is made available as the root of an FTP server; however, if a special folder type is selected, then a different folder will be made available as the root of the FTP server that this folder configures.

Indicates the type of FTP folder made available. Specify one of the following:

- `Default (0)` - Default. The folder that is made available as the root of the FTP server. This folder is a normal FTP folder, behaving the same way as a folder or page in the My

webMethods Server taxonomy. The Ftp Folder physically stores files published to the FTP server in this FTP folder taxonomy.

- `User Home Folder (1)` - Makes the Home Folder of the My webMethods Server user who is publishing data (also known as My Folders) available as the root of the FTP server. When a user logs on to the FTP server, My webMethods Server automatically uses that user's Home Folder, which means that each user can only access his or her own Home Folder.
- `Public Folders (2)` - Makes the My webMethods Server Public Folders available as the root of the FTP server. Specify `Public Folders` to provide quick FTP access to Public Folders and the taxonomy under it.
- `Root Folder (3)` - Makes the My webMethods Server Root Folder available as the root of the FTP server. The Root Folder is the root of all folders, and the parent of Public Folders, the System folder, and so on. Specify `Root Folder` to provide quick FTP access to the entire taxonomy.

Http Header Policy DBO

Portlet Title	Http Header Policy
Portlet Name	wm_xt_httpheaderpolicy
Portlet File Name	wm_xt_httpheaderpolicy.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	Folders > System > Policy Providers > Available Types > Http Header Policy Provider

Security. Administrators use this security provider to configure permissions on My webMethods Server items. Unlike other security providers where privileges are manually assigned, use this security provider to dynamically grant or deny access to server objects based on information that is passed to My webMethods Server in the HTTP header when a user logs in.

Administrators select to use the HTTP Header Policy security provider when creating a new security realm. This security provider is useful when My webMethods Server is protected by a front-end authentication product, for example, SiteMinder.

The HTTP Header Policy security provider grants or denies READ, MODIFY, CREATE CHILD, SET PERMISSIONS, and DELETE privileges. To grant/deny privileges, the security provider first examines the key/value pairs in the HTTP header attempting to find those that match key/value pairs configured in the properties of the HTTP Header Policy portlet. Administrator can configure key/value pairs for each of the privileges. When the security provider finds a matching key/value pair in the HTTP header, it determines whether to grant or deny the privilege based on the value of the Authorization Policy (authorizationPolicy) property.

If an administrator does not configure a key/value pair for a privilege, the security provider uses the Default Right Set (defaultRightSet) property to determine whether to grant or deny that privilege. For example, if an administrator does not configure a key/value pair for the CREATE CHILD privilege, the security provider grants the privilege if it is configured in the default right set or denies the privilege if it is not configured in the default right set.

Policy Configuration Properties

Default Right Set (defaultRightSet)
Identifies the set of default privileges to grant to the security realm. The security provider uses the default right set to determine whether to grant a privilege when there is no configured key/value pair for that privilege, or when the configured key/value pair is not present in the HTTP header. By default, READ is included in the default right set.

When using the portlet's user interface, select the check boxes that correspond to the privileges that you to include in the default right set. When programmatically configuring this portlet, provide a bit field to indicate the privileges to include in the default right set. The following table shows the bits associated with each privilege:

Bits	Privileges
`00000001`	READ
`00000100`	CREATE CHILD
`00010000`	MODIFY
`01000000`	SET PERMISSIONS
`10000000`	DELETE

For example, to set the default right set to include the READ, SET PERMISSIONS, and DELETE privileges, specify the bit field `11000001`.

Authorization Policy (authorizationPolicy)
Indicates whether to grant or deny the privileges associated with the configured key/value pairs found in the HTTP header. For example, if the security provider finds the key/value pair identified by the READ Header (readHeader) property, it grants or denies READ access based on the value of this property. Specify one of the following:

- `GRANT` – Default. Grant the privileges associated with the configured key/value pairs that are found in the HTTP header.
- `DENY` – Deny the privileges associated with the configured key/value pairs that are found in the HTTP header.

READ Header (readHeader)
Identifies the key/value pair in the HTTP header to associate with the READ privilege. If the security provider finds the key/value pair in the HTTP header, it grants or denies the privilege based on the value of the Authorization Policy (authorizationPolicy) property.

For example, you might set this property to `read_privilege=UseSecurityProvider` and the Authorization Policy (authorizationPolicy) to `GRANT`. If the security provider finds the key/value pair `read_privilege=UseSecurityProvider`in the HTTP header, it grants the READ privilege.

If the property has no value, the security provider grants the READ privilege if it is included in the Default Right Set (defaultRightSet) property or denies the READ privilege if it is not included in the Default Right Set (defaultRightSet) property.

CREATE CHILD Header (createItemHeader)
Identifies the key/value pair in the HTTP header to associate with the CREATE CHILD privilege. The CREATE CHILD privilege controls whether the user can create new items or create subfolders in folders controlled by the security realm. If the security provider finds the key/value pair in the HTTP header, it grants or denies the privilege based on the value of the Authorization Policy (authorizationPolicy) property.

For example, you might set this property to `create_privilege=UseSecurityProvider` and the Authorization Policy to `GRANT`. If the security provider finds the key/value pair `create_privilege=UseSecurityProvider` in the HTTP header, it grants the CREATE CHILD privilege.

If the property has no value, the security provider grants the CREATE CHILD privilege if it is included in the Default Right Set (defaultRightSet) property or denies the CREATE CHILD privilege if it is not included in the Default Right Set (defaultRightSet) property.

MODIFY Header (modifyHeader)
Identifies the key/value pair in the HTTP header to associate with the MODIFY privilege. If the security provider finds the key/value pair in the HTTP header, it grants or denies the privilege based on the value of the Authorization Policy (authorizationPolicy) property.

For example, you might set this property to `modify_privilege=UseSecurityProvider` and the Authorization Policy to `DENY`. If the security provider finds the key/value pair `modify_privilege=UseSecurityProvider` in the HTTP header, it denies the MODIFY privilege.

If the property has no value, the security provider grants the MODIFY privilege if it is included in the Default Right Set (defaultRightSet) property or denies the MODIFY privilege if it is not included in the Default Right Set (defaultRightSet) property.

SET PERMISSIONS Header (setPermissionsHeader)
Identifies the key/value pair in the HTTP header to associate with the SET PERMISSIONS privilege. The SET PERMISSIONS privilege controls whether the user can modify the permissions of items controlled by the security realm. If the security provider finds the key/value pair in the HTTP header, it grants or denies the privilege based on the value of the Authorization Policy (authorizationPolicy) property.

For example, you might set this property to `permissions_privilege=UseSecurityProvider` and the Authorization Policy to `GRANT`. If the security provider finds the key/value pair `permissions_privilege=UseSecurityProvider` in the HTTP header, it grants the SET PERMISSIONS privilege.

If the property has no value, the security provider grants the SET PERMISSIONS privilege if it is included in the Default Right Set (defaultRightSet) property or denies the SET PERMISSIONS privilege if it is not included in the Default Right Set (defaultRightSet) property.

DELETE Header (deleteHeader)

Identifies the key/value pair in the HTTP header to associate with the DELETE privilege. If the security provider finds the key/value pair in the HTTP header, it grants or denies the privilege based on the value of the Authorization Policy (authorizationPolicy) property.

For example, you might set this property to `delete_privilege=UseSecurityProvider` and the Authorization Policy (authorizationPolicy) to `DENY`. If the security provider finds the key/value pair `delete_privilege=UseSecurityProvider`in the HTTP header, it denies the DELETE privilege.

If the property has no value, the security provider grants the DELETE privilege if it is included in the Default Right Set (defaultRightSet) property or denies the DELETE privilege if it is not included in the Default Right Set (defaultRightSet) property.

Locale Property Storage DBO

Portlet Title	Locale Property Storage
Portlet Name	wm_xt_localepropstorage
Portlet File Name	wm_xt_localepropstorage.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_xt_localepropstorage
Default Instances of the portlet	None

Dynamic Business Objects. The system uses this system dynamic business object (DBO) to implement multi-locale property value storage. When implemented for a portlet, an end user can define different property values based on locales. My webMethods Server automatically publishes instances of this Locale Property Storage DBO; users cannot publish it.

An example of a portlet that uses this support is the HTML Text Portlet, which allows per-locale storage for the Text (HTMLText) property. End users can define different HTML text to display for one or more locales. The portlet stores the property value for each locale and returns the appropriate value based on the end user's current locale.

Mime Message DBO

Portlet Title	Mime Message
Portlet Name	wm_xt_mimemessage
Portlet File Name	wm_xt_mimemessage.pdp
Top-level Folder	extras
JSR168 Portlet?	No

Alias	None
Default Instances of the portlet	None

Dynamic Business Objects. The system uses an instance of this data type in My webMethods Server to publish an e-mail message that was retrieved from an e-mail inbox or NNTP newsgroup on an external server to a file in a folder. My webMethods Server typically uses this data type when synchronizing an e-mail inbox or NNTP newsgroup to a wm_xt_nntpfolder and wm_xt_pop3folder folder instance.

This data type is a My webMethods Server custom data type, extending the "File" data type to make available the additional e-mail related properties that are embedded in e-mail files. When an e-mail message is published to a folder, the message is parsed and the extended properties are extracted and saved for easy subsequent access.

General Properties

Name (name)
Required. Identifies the folder where the messages are to be published. Specify the name of the folder.

Description (description)
Provides a the description of the folder where the messages are to be published. If the property has no value, the folder will not have a description.

Keywords (keywords)
Defines the keywords for the folder where the messages are to be published. These keywords can be associated with items in the folder and subsequently searched. If the property has no value, the folder will not have keywords associated with it.

Location Properties

Parent Folder (parentID)
Required. Identifies the parent folder of the folder where the messages are to be published. Specify the ID of the parent folder.

File Properties

File (origFileName)
Required. Identifies the e-mail file to publish. Specify the original file name of the e-mail file. During publishing, this field also holds the uploaded file. If this property has no value during publishing, the creation of the instance fails.

Message Properties

Subject (msgSubject)
Required. Holds the subject of the e-mail file being published. The portlet automatically extracts the subject from the file during publishing. You do not need to provide a value.

From (msgFrom)
Required. Holds the "From" e-mail address from the e-mail file being published. The portlet automatically extracts the e-mail address from the file during publishing. You do not need to provide a value.

Sent (msgSent)
Required. Holds the sent date from the e-mail file being published. The portlet automatically extracts the date from the file during publishing. You do not need to provide a value.

Size (msgSize)
Required. Holds the size of the published message. The portlet automatically extracts the size during publishing. You do not need to provide a value.

My webMethods Folder DBO

Portlet Title	My webMethods Folder
Portlet Name	wm_xt_fabricfolder
Portlet File Name	wm_xt_fabricfolder.pdp
Top-level Folder	None
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	There are hundreds of these folders inside the My webMethods Applications Taxonomy, depending on how many "Fabric UIs" are deployed (for example, Monitor, Optimize, and so on.)

Dynamic Business Objects. Page developers use this dynamic business object to provide additional data to folders such as whether it is an "isTaskFolder".

One special behavior of this folder is that whenever a child folder is created that child is automatically converted from type folder to type "wm_xt_fabricfolder".

The Main Nav Portlet pays special attention to folders of type "wm_xt_fabricfolder". If a Fabric Nav portlet is showing navigation to folders of type "wm_xt_fabricfolder", it will only draw links if the isTaskFolder property is set to true.

Additionally, whenever a "wm_xt_fabricfolder" is created, if it has the "isTaskFolder" property set to true, it will automatically generate a Security Realm and add itself to that Security Realm.

Properties

Is Openable (isOpenable)
Provides a true/false flag to indicate whether this folder is openable when displayed in the LHS navigation tree. If set to true, the folder will open as a new tab when clicked. If set to false, it will act as a simple container node in the LHS and will only expand/contract. If this property has no value, it defaults to true.

Is Task Folder (isTaskFolder)
Determines whether or not this Fabric Folder should generate a Security Realm on creation and whether or not the Main Nav Portlet should be allowed to show this folder in its navigation. If this property has no value, the property defaults to false and this folder acts as a regular, basic folder.

Required Products (requiredProducts)
Deprecated.

Keywords (keywords)
Provides a keywords list assigned to this object. This property can be used for searches in the My webMethods Server taxonomy. If this property has no value, no keywords will be displayed for this object.

Parent Folder (parentID)
Required. Used during publishing through the Publish portlet. It defines the parent folder ID where this object is stored.

NNTP Folder DBO

Portlet Title	NNTP Folder
Portlet Name	wm_xt_nntpfolder
Portlet File Name	wm_xt_nntpfolder.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Containers. Administrators publish an instance of this folder type to a My webMethods Server to provide a way to view the contents of an NNTP newsgroup. The NNTP Folder polls an NNTP newsgroup and creates entries within the folder for each newsgroup message it retrieves.

Typically, My webMethods Server uses the Mime Message Folder Renderer Portlet or NNTP Folder Threaded View Portlet to render an instance of a NNTP Folder.

- The Mime Message Folder Renderer Portlet displays the list of newsgroup messages as a table.
- The NNTP Folder Threaded View Portlet displays the list of newsgroup messages as a set of threads

A user can click a newsgroup message to view the contents of that message.

The NNTP Folder also adds a **Fetch New Articles** option to the Tools menu so that end users can force an immediate refresh of messages from the NNTP newsgroup, overriding the refresh interval.

Extended Properties

Refresh Interval (refreshInterval)
Indicates how many hours you want the portlet to wait before polling the NNTP newsgroup again to check for new newsgroup messages. Specify 1, 2, 4, 5, 12, or 24 hours. If the property has no value, for example, when set to an empty value, then the portlet continuously polls the mailbox.

Connection Parameters

Host (host)
Identifies the NNTP server for the NNTP newsgroup. Specify the host name of the server. If the property has no value, the portlet does not make attempts to retrieve newsgroup messages and, as a result, the NNTP Folder will contain no content.

Port (port)
Identifies the port to use to connect to the NNTP server. Specify the port number. The standard NNTP server port number is 119. If the property has no value, for example, when set to an empty value, then the portlet does not make attempts to retrieve newsgroup messages and, as a result, the NNTP Folder will contain no content.

User Name (userName)
Identifies the user name the portlet is to supply to the NNTP server for authentication. Specify a valid user name for the NNTP server. If the property has no value, the portlet cannot retrieve messages and, as a result, the NNTP Folder will contain no content.

Password (password)
Provides the password for the user name. The NNTP Folder portlet supplies this password, along with the user name, to the NNTP server for authentication. If the property has no value, the portlet cannot retrieve messages and, as a result, the NNTP Folder will contain no content.

NewsGroup Name (newsGroupName)
Identifies the newsgroup to poll. Specify the name of the newsgroup. If the property has no value, the portlet will retrieve no messages and, as a result, the NNTP Folder will contain no content.

Pop3 Folder DBO

Portlet Title	Pop3 Folder
Portlet Name	wm_xt_pop3folder
Portlet File Name	wm_xt_pop3folder.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	None

Default Instances of the portlet None

Containers. Administrators publish an instance of this folder type to a My webMethods Server page to provide a way to view the contents of a POP3 mailbox. The Pop3 Folder polls a POP3 mailbox and creates entries within the folder for each e-mail message it retrieves.

By default, My webMethods Server uses the Mime Message Folder Renderer Portlet to render an instance of a Pop3 Folder. The Mime Message Folder Renderer Portlet renders the mailbox content by displaying the list of e-mail messages in a table. A user can click an e-mail message in the table to view the contents of that message.

The Pop3 Folder also adds a **Fetch New Messages** option to the Tools menu so that end users can force an immediate refresh of messages from the POP3 mailbox, overriding the refresh interval.

Extended Properties

Refresh Interval (refreshInterval)
Indicates how many hours you want the portlet to wait before polling the POP3 mailbox again to check for new e-mail messages. Specify 1, 2, 4, 5, 12, or 24 hours. If the property has no value, for example, when set to an empty value, then the portlet continuously polls the mailbox.

Connection Parameters

Host (host)
Identifies the POP3 server for the POP3 mailbox. Specify the host name of the server. If the property has no value, the portlet does not make attempts to retrieve e-mail messages and, as a result, the Pop3 Folder will contain no content.

Port (port)
Identifies the port to use to connect to the POP3 server. Specify the port number. The standard POP3 server port number is 30. If the property has no value, the portlet does not make attempts to retrieve e-mail messages and, as a result, the Pop3 Folder will contain no content.

User Name (username)
Identifies the mailbox on the POP3 server from which to retrieve messages. Specify the user name associated with the mailbox. The Pop3 Folder supplies this user name to the POP3 server for authentication. If the property has no value, the portlet cannot retrieve e-mail messages and, as a result, the Pop3 Folder will contain no content.

Password (password)
Provides the password for the user name. The Pop3 Folder supplies this password, along with the user name, to the POP3 server for authentication. If the property has no value, the portlet cannot retrieve e-mail messages and, as a result, the Pop3 Folder will contain no content.

Portal Skin DBO

Portlet Title Portal Skin

Portlet Name	wm_xt_skin
Portlet File Name	wm_xt_skin.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/xtype/wm_xt_skin
Default Instances of the portlet	None

Dynamic Business Objects. Page developers use this object type to define a new skin and configure its properties.

All skins that a page developer creates by using the Administration Dashboard > User Interface > Skin Administration portlet are instances of this object type. For detailed information about creating skins, see information about skins in *Administering My webMethods Server*.

Properties

Style (visualStyle)
Identifies the JSP file to us to render the skin's properties. Do not change this value. All instances of the portlet share the value of this property.

General Properties

Name (name)
Required. Defines the name of the skin. This is an internal name and is not displayed to the end user. Specify an alphanumeric string that contains less than 100 characters. You can use ASCII letters, numbers, and the underbar character.

Description (description)
Defines a description for the skin. This is the display name of the skin that the end user sees. Specify a string that contains less than 255 characters. If the property has no value, the skin will not have a description.

Keywords (keywords)
Defines one or more keywords (or tags) to use when searching for skins. Specify a comma-separated list of keywords. If the property has no value, the skin will not have any keywords assigned to it.

Location Properties

Parent Folder (parentID)
Required. Identifies the folder where the skin you are defining is stored. Specify the thingID or alias of the folder. This is usually `folder.skins`.

Skin Properties

Server Side Resource Path (serverSideResourcePath)
Required. Defines the base location for skin resources that are used on the server side, for example, skin properties file. Specify the location as a path within the portal.war; that is, specify `/ui/skins/` *skin_name*, where *skin_name* is the name of the skin.

Client Side Resource Path (clientSideResourcePath)
Required. Defines the base URL for skin resources that are used on the client side, for example, the CSS and image files. Specify the base URL as a path from the server root; that is, specify `/ui/skins/` *skin_name*, where *skin_name* is the name of the skin.

CSS Preview (cssPreview)
Required. Defines the CSS style declaration to use on a menu item displaying the name of the skin. Specify a name that indicates what the skin looks like. For example, for a skin with a blue and red color scheme, you might specify:

```
color: blue; background-color: red;
```

Relative Link DBO

Portlet Title	Relative Link
Portlet Name	wm_xt_relativelink
Portlet File Name	wm_xt_relativelink.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Dynamic Business Objects. Page developers use instances of this type to add links that are relative to the front-end server to a page. To provide a set of links for the end user, page developers publish multiple instances of this link type to a folder.

When publishing an instance of this link, developers specify a URL relative to the server root, such as "/folder.public" or "/folder.system?layout=details", and so on. When rendered, the link will be automatically prepended with the front-end URL of the My webMethods Server cluster, for example, "http://example.com/folder.public" or "http://mws.example.com:8585/folder.system?layout=details".

Link Properties

URL (URL)

Required. Defines the link. Specify the link path relative to the My webMethods Server front-end URL.

Link Target (linkTarget)
Identifies the window or frame in which the link should open. Specify one of the following:

- `null` – Also blank (""). Default. When the property has no value, the link opens in the current window.
- `_self` – The link opens in the current window.
- `_blank` – The link opens in a new, unique window.
- `portal-link` – The link opens in a new, shared window.

Restful Link DBO

Portlet Title	Restful Link
Portlet Name	wm_xt_restfullink
Portlet File Name	wm_xt_restfullink.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_xt_restfullink
Default Instances of the portlet	None

Dynamic Business Objects. Page developers use this portlet to build a link to another My webMethods Server page. In addition to linking to another page, this portlet also allows the page developer to supply initial values of properties for a portlet on the target page.

Link Properties

Base Page ID (baseID)
Required. Identifies the target

Item ID (uriID)
Identifies the portlet on the target page for which to supply property values. Specify the alias or thingID of the portlet. If the property has no value, the link behaves like a regular link; property values will not be supplied for a target portlet.

Parameters (linkParams)
Defines the property values to use for the target portlet. Specify a comma-separated list of property name/value pairs. Each pair in the list should use the format:

name=value

If the property has no value, no property values are assigned to the target portlet.

Role DBO

Portlet Title	Role
Portlet Name	wm_xt_role
Portlet File Name	wm_xt_role.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Dynamic Business Objects. Administrators use this dynamic business object (DBO) in My webMethods Server to store general role properties for a role. My webMethods Server implicitly creates instances of this DBO when new roles are added to the system.

Users should never publish instances of this DBO. However, administrators can use this DBO to create roles from xmlImport.xml script files, for example, to create multiple roles to be deployed to multiple servers. The following is an example of an xmlImport script file that creates a static role named "Test Role". "Test Role" has the members: "user1", "user2", and "user3".

```
<context alias="role.provider.storage">
   <!-Creating storage for static role -->
   <wm_xt_staticrole name="Test Role" principals="user1,user2,user3"
      alias="test. role.storage" targetURI="test.role.storage" />
</context>
<context alias="role.provider.roles">
   <!-Creating Role instance -->
   <wm_xt_role name="Test Role"
      roleProviderURI="role.static.provider"
      alias="test.role">
   <relation path="test.role.storage"
      type="role.relation"
   />
</context>
```

General Properties

Name (name)
Required. Defines the role name. Specify a name that is unique among all role names.

Description (description)
Defines a description of the role. Specify a description. If the property has no value, the role will not have a description.

Keywords (keywords)

Defines keywords for the role. Specify keywords that can be used when searching for the role in the My webMethods Server taxonomy. If the property has no value, no keywords are assigned to the role.

Parent Folder (parentID)
Required. Identifies where to store the role. Specify the parent folder ID.

Role Properties

Role Provider (roleProviderURI)
Required. Defines the alias or thingID of the role type provider to use for this instance. Specify one of the following:

- `role.static.provider` – Default. Static role.
- `role.ldap.query.provider` – LDAP query role.
- `role.db.provider` – Database role.
- `role.rule.provider` – Rule-based role.

Shell DBO

Portlet Title	Shell
Portlet Name	wm_xt_shell
Portlet File Name	wm_xt_shell.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/xtype/wm_xt_shell
Default Instances of the portlet	None

Dynamic Business Objects. Page developers use this object type to define a new shell and configure its header, footer, left navigation, and right navigation.

All shells that a page developer creates by using the Administration Dashboard > User Interface > Shell Administration portlet are instances of this object type. For detailed information about creating shells, see *Administering My webMethods Server*.

Properties

Visual Style (visualStyle)
Identifies the JSP file to use to render the shell's properties. Do not change this value. The default value is `dbo_view.jsp`. All instances of the portlet share the value of this property.

General Properties

Name (name)
Required. Defines the name for the shell. Specify a string that contains less than 100 characters.

Description (description)
Defines a description for shell. Specify a string that contains less than 255 characters. If the property has no value, the shell will not have a description.

Keywords (keywords)
Defines one or more keywords (or tags) to use when searching for shells. Specify a comma-separated list of keywords. If the property has no value, the shell will not have any keywords assigned to it.

Location Properties

Parent Folder (parentID)
Required. Identifies the folder where the shell you are defining is stored. Specify the thingID or alias of the folder. This is usually `folder.shells`.

Shell Properties

Parent (parent)
Defines the parent shell to use to obtain property values for properties not explicitly defined for the shell you are creating. The shell you create will inherit shell-category properties from the parent shell. If the property has no value, the shell you are defining will not inherit values.

Header (head)
Defines the header to use for the shell. Specify the thingID or alias of the page to use as the shell header. If the property has no value, the shell inherits the header from the parent, or if you did not specify the Parent (parent) property, the shell will not have a header.

Footer (footer)
Defines the footer to use for the shell. Specify the thingID or alias of the page to use as the shell footer. If the property has no value, the shell inherits the footer from the parent, or if you did not specify the Parent (parent) property, the shell will not have a footer.

Leftnav (leftnav)
Defines the left navigation to use for the shell. Specify the thingID or alias of the page to use as the left navigation. If the property has no value, the shell inherits the left navigation from the parent, or if you did not specify the Parent (parent) property, the shell will not have left navigation.

Rightnav (rightnav)
Defines the right navigation to use for the shell. Specify the thingID or alias of the page to use as the right navigation. If the property has no value, the shell inherits the right navigation from the parent, or if you did not specify the Parent (parent) property, the shell will not have right navigation.

Titlebar (titlebar)

Defines the title bar to use for the shell. Specify the thingID or alias of the page to use as the title bar. If the property has no value, the shell inherits title bar from the parent, or if you did not specify the Parent (parent) property, the shell will not have a title bar.

Shortcut DBO

Portlet Title	Shortcut
Portlet Name	wm_xt_shortcut
Portlet File Name	wm_xt_shortcut.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Dynamic Business Objects. Page developers use instances of this type to define shortcuts to My webMethods Server resources.

Properties

Display Style (visualStyle)
Identifies the JSP file to use to render the shortcut's properties. The default value is `dbo_view.jsp`.

General Properties

Description (description)
Defines a description of this shortcut. Must be less than 255 characters. If the property has no value, no description will be displayed for this shortcut.

Name (name)
Required. Defines a name for this shortcut. Must be less than 100 characters.

Keywords (keywords)
Defines the keywords to associate with this shortcut. The total number of characters for all keywords combined must be less than 255 characters. If this property has no value, no keywords will be associated with this shortcut.

Location Properties

Parent Folder (parentID)
Required. Identifies the folder ID that contains this shortcut. Specify the thingID or alias of a folder.

Link Properties

Target Resource (URL)
Required. Identifies the target of this shortcut. Specify the thingID or alias of a resource.

Link Target (linkTarget)
Defines the window or frame into which the target resource should open when the user clicks the shortcut link. Possible values include:

- `blank` – Also empty (""). Link opens in current window.
- `_blank` – Link opens in a unique window.
- `specified-name` – Link opens in a frame or window with the specified name, that is, (specified-name).

Target Db ID (targetDbID)
Do not modify. Used internally.

Target XType (targetXType>)
Do not modify. Used internally.

Target Data Source (targetDataSource)
Do not modify. Used internally.

Static Role Provider DBO

Portlet Title	Static Role Provider
Portlet Name	wm_xt_staticrole
Portlet File Name	wm_xt_staticrole.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Dynamic Business Objects. Administrators use this dynamic business object (DBO) in My webMethods Server to store configuration for static roles. My webMethods Server implicitly creates instances of this DBO when new static roles are added to the system.

Users should never publish instances of this DBO. However, administrators can use this DBO to create static roles from xmlImport.xml script files, for example, to create multiple static roles to be deployed to multiple servers. The following is an example of an xmlImport script file that creates a static role named "Test Role". "Test Role" has the members: "user1", "user2", and "user3".

```
<context alias="role.provider.storage">
```

```
   <!-Creating storage for static role -->
   alias="test.role.storage"
</context>
<context alias="role.provider.roles">
   <!-Creating Role instance -->
   <wm_xt_role name="Test Role" roleProviderURI="role.static.provider"
      alias="test. role"> <relation path="test.role.storage"
      type="role.relation"
   />
</context>
```

General Properties

Name (name)
Required. Defines a name for the static role. Specify a name that is unique among all role names.

Description (description)
Defines a description for the static role. Specify a description. If the property has no value, the role will not have a description.

Keywords (keywords)
Defines keywords for the static role. Specify keywords that can be used when searching for the static role in the My webMethods Server taxonomy. If the property has no value, no keywords are assigned to the role.

Parent Folder (parentID)
Required. Identifies where to store the role. Specify the parent folder ID.

Role Membership Properties

Principals (principals)
Required. Defines the members of the static role. Specify a string that contains a comma-separated list of the principal UIDs.

Website Archive Viewer DBO

Portlet Title	Website Archive Viewer
Portlet Name	wm_xt_websitearchive
Portlet File Name	wm_xt_websitearchive.pdp
Top-level Folder	extras
JSR168 Portlet?	No
Alias	/xtype/wm_xt_websitearchive
Default Instances of the portlet	None

Page components. Page developers use this File dynamic business object (DBO) to upload a static web site archive (either a .zip or .jar), and to publish the web site archive so that end users can browse its contents within the My webMethods Server taxonomy.

The Website Archive Viewer DBO extends the basic file DBO functionality (upload, download, etc.) with special rendering features. Uploaded files must be in .zip or .jar format and contain a collection of static web resources, for example, HTML pages, image files, etc. A page developer publishes the uploaded file using the Publish Portlet; the Page Editor cannot be used. After the file is published, when the end user clicks on the file, the system displays the contents of the web site archive in an IFrame. The page developer configures the Website Archive Viewer DBO to define the initial page to display during the publish process.

File Properties

WebSite Archive (origFileName)
Required. Identifies the archive to upload. Use the user interface to select a local file on your machine to upload. The file extension of the file must be either .zip or .jar. This property value is not shared among portlet instances. Instances of the portlet do not share the value of this property.

Web Site Properties

Default Page (defaultPage)
Required. Identifies the file within the uploaded web site archive to use as the starting page to display for the web site. The default is `index.html`. Instances of the portlet do not share the value of this property.

Frame Height (frameHeight)
Defines the height of the IFrame to use to display the web site. Specify the number of pixels. The default is `400`. Instances of the portlet do not share the value of this property.

5 Item Editing Tools

This chapter describes the portlets and components that My webMethods Server provides out-of-the-box for item editing tools.

Summary of Portlets and Components

The following table lists the portlets and components that My webMethods Server provides out-of-the-box for item editing tools.

Name	Description
Comments Page Portlet	System. Page developers and end users use this portlet to associate comment text with a My webMethods Server page, folder, or portlet. The Comments Page portlet displays existing comments and provides an editor for adding additional comments. Each comment includes the comment text, the date, and the username of the user who entered the comment.
Dynamic Business Object Viewer Portlet	System. Page developers use this portlet to render a custom user interface for dynamic business object (DBO) instances on a My webMethods Server page. Using this portlet is a limited and deprecated way to provide a custom user interfaces for custom types (xtypes).
Properties Page Portlet	System. Administrators, page developers, and end users use this portlet to view and edit My webMethods Server object properties.
Sort Order Portlet	Page Authoring. End users use this portlet to specify a custom sort order for items in a folder. Page developers can configure this portlet to override the default URLs to display when the end user clicks the portlet's **Apply** and **Cancel** buttons.

Comments Page Portlet

Portlet Title	Comments Page
Portlet Name	wm_comments
Portlet File Name	wm_comments.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.comments
Default Instances of the portlet	None

System. Page developers and end users use this portlet to associate comment text with a My webMethods Server page, folder, or portlet. The Comments Page portlet displays existing comments

and provides an editor for adding additional comments. Each comment includes the comment text, the date, and the username of the user who entered the comment.

The Comments Page portlet is included in the default shell in the page or portlet tool bar as the Comment menu item.

Page developers can publish an instance of the Comments Page portlet to any My webMethods Server or folder.

Properties

Item (commentTargetURI)
Identifies the My webMethods Server object for which to attach a comment. Specify the URI or alias of the object. If the property has no value or when the portlet is used from the tool bar menu, the portlet attaches the comment to the current resource.

Return URL (returnUrl)
Identifies the page to display when the end user clicks the Done button of the Comments Page portlet. Specify the URL or alias of the page to display. If the property has no value, the Comments Page portlet displays the previously viewed My webMethods Server page.

Dynamic Business Object Viewer Portlet

Portlet Title	Dynamic Business Object Viewer
Portlet Name	wm_xtypeview
Portlet File Name	wm_xtypeview.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/m_xtypeview
Default Instances of the portlet	None

System. Page developers use this portlet to render a custom user interface for dynamic business object (DBO) instances on a My webMethods Server page. Using this portlet is a limited and deprecated way to provide a custom user interfaces for custom types (xtypes).

The four base types of DBOs are: Folders, Forms, Content (Files), and Links. When an instance of one of these base types or one of their xtypes is dropped onto a page (folder), by default, My webMethods Server uses a default renderer for the instance. However, some xtypes might provide a custom user interface. Use this Dynamic Business Object Viewer portlet to initiate the custom user interface.

Properties

DBO Instance (xthing)

Required. Identifies the xtype instance to display. Specify the URI of the xtype. All instances of the portlet share the value of this property.

Properties Page Portlet

Portlet Title	Properties Page
Portlet Name	wm_properties
Portlet File Name	wm_properties.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_properties
	portlet.properties
Default Instances of the portlet	Under system folder.portlets folder

System. Administrators, page developers, and end users use this portlet to view and edit My webMethods Server object properties.

Page developers can publish this portlet to custom pages and configure it to identify the My webMethods Server object for which to display properties so that users can view/edit those properties.

In addition, My webMethods Server displays this portlet when a user selects the Properties menu for any object.

Properties

targetURI (targetURI)
Required. Identifies the My webMethods Server object for which to display properties. Specify the URI of the object. Instances of the portlet do not share the value of this property.

Sort Order Portlet

Portlet Title	Sort Order
Portlet Name	wm_sortorder
Portlet File Name	wm_sortorder.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.sortorder

Default Instances of the portlet None

Page Authoring. End users use this portlet to specify a custom sort order for items in a folder. Page developers can configure this portlet to override the default URLs to display when the end user clicks the portlet's **Apply** and **Cancel** buttons.

Data Properties

Target Folder (target)
Required. Identifies the folder to which the custom sort order applies. Specify the thingID or alias of a My webMethods Server folder.

Behavior Properties

finishUrl (_finishUrl)
Identifies the page to display when the end user clicks the Apply button. Specify the URL of the page. By default, the property is set to the URL of the previous page.

Cancel URL (_cancelUrl)
Identifies the page to display when the end user clicks the Cancel button. Specify the URL of the page. By default, the property is set to the URL of the previous page.

6 Developer Tools

This chapter describes the portlets and components that My webMethods Server provides out-of-the-box for developer tools.

Summary of Portlets and Components

The following table lists the portlets and components that My webMethods Server provides out-of-the-box for developer tools.

Name	Description
SOAP Monitor Portlet	Tools. Page developers use this portlet to debug issues with web service calls from other portlets. The portlet captures the SOAP request and response messages for calls from the other portlets.
Style Sheet Portlet	Tools. Page developers can use this portlet to test skins they are developing. The portlet includes common portlet styles, including HTML elements that contain JSR168 portlet CSS classes, and a few common constructs like tabs, property groups, and tables. As a result, page developers can display this portlet using the skin they are developing to view how the skin looks with common portal content.

SOAP Monitor Portlet

Portlet Title	SOAP Monitor
Portlet Name	wm_mws_diagnostics___soapmonitor
Portlet File Name	wm_mws_diagnostics.war
Top-level Folder	admin
JSR168 Portlet?	No
Alias	portlet.soapmonitor
Default Instances of the portlet	Folders > Administrative Folders > Administration Dashboard> Analysis > SOAP Monitor

Tools. Page developers use this portlet to debug issues with web service calls from other portlets. The portlet captures the SOAP request and response messages for calls from the other portlets.

Style Sheet Portlet

Portlet Title	Style Sheet
Portlet Name	wm_stylesheet

Portlet File Name	wm_stylesheet.pdp
Top-level Folder	development
JSR168 Portlet?	No
Alias	/portlet/wm_stylesheet
Default Instances of the portlet	None

Tools. Page developers can use this portlet to test skins they are developing. The portlet includes common portlet styles, including HTML elements that contain JSR168 portlet CSS classes, and a few common constructs like tabs, property groups, and tables. As a result, page developers can display this portlet using the skin they are developing to view how the skin looks with common portal content.

7 Legacy Portlets and Backwards Compatibility

This chapter describes the portlets, components, and dynamic business objects (DBOs) that My webMethods Server provides out-of-the-box for working with legacy portlets and components or for backwards compatibility.

Summary of Portlets, Components, and DBOs

The following table lists the portlets and components that My webMethods Server provides out-of-the-box for working with legacy portlets and components or for backwards compatibility.

Name	Description
Advanced Tooltip Portlet	UI Controls. Page developers use this portlet to provide an advanced tooltip for any HTML element in a page.
Calendar Picker Portlet	System. Page developers who are building 6.x proprietary, non-JSR168 portlets use this portlet to add the Date property editor to a page. The Date property editor renders a text input file with a calendar icon next to it. When an end user clicks either the input field or the icon, the Calendar Picker portlet displays a calendar popup window from which the end user can select a date. After selecting the date and closing the popup window, the portlet displays the selected date in the input field.
Dynamic Table Portlet	System. Page developers using My webMethods Server version 6.x can use this portlet to display data in a table. For My webMethods Server version 7.x and later, use the Table controls that the webMethods CAF framework provides.
Dynamic Tree Portlet	System. Page developers using My webMethods Server version 6.x can use this portlet to display data in a tree. For My webMethods Server version 7.x and later, use the Tree controls that the webMethods CAF framework provides.
Dynamic Tabs Portlet	System. Page developers using My webMethods Server version 6.x can use this portlet to display a configured list of tabs. For My webMethods Server version 7.x and later, use the Tab controls that the webMethods CAF framework provides.
Extended Property Editors Component	Property Editors. Page developers use this component to provide specialized user interfaces for editing portlet preferences on the Portlet Properties page. This component is a collection of custom editors that other portlets can reuse.
I18N Property Editors Component	Property Editors. Page developers use this component to provide specialized user interfaces for editing internationalization-specific portlet preferences on the Portlet Properties page. This component is a collection of custom editors that other portlets can reuse.
Legacy Property Editors Component	Property Editors. Portlet developers use this component to provide specialized user interfaces for editing portlet preferences

Name	Description
	on the Portlet Properties page. This component is a collection of legacy property editors that are provided for backwards compatibility with earlier releases of My webMethods Server. They map old property editors to the new property editors that provide the equivalent functionality.
The Machine Skin	Skins. This component is a skin that is installed, but not used by default. Page developers can use this skin as the default skin or the target of a Skin Rule. Additionally, page developers can use the skin as the base for a custom skin.
PCA Submit Button Portlet	Property Editors. Page developers use this component to provide specialized legacy Portlet-Controller Architecture (PCA) controls. This component is a collection of custom editors that other legacy portlets can reuse.
Popup Tab Styles Portlet	Page Components. Page developers indirectly use this portlet when using the Dynamic Tabs Portlet (wm_tabs) to display popup style tabs. The Popup Tab Styles Portlet displays target text, and as the end user clicks or hovers over the target text, the portlet displays its list of tabs as a popup menu.
Progress Indicator Portlet	UI Controls. This is a deprecated portlet. The portlet displays a user interface control that shows a progress animation. The portlet was used as part of the legacy common search framework. Some existing portlets that have not been ported to webMethods CAF still use this portlet.
Resource List View Portlet	Template. Page developers use this portlet in a shell section, portal page, or legacy portlet to display a list of items in a table format. The portlet retrieves the My webMethods Server items to display in the table by executing a My webMethods Server command. The page developer configures the columns to display in the table.
Selected List View Portlet	Template. Page developers use this portlet to display tabs that represent the folders in a hierarchy. The page developer identifies the folders for which to display tabs by specifying the root folder of the hierarchy. At run time, the portlet dynamically determines the list of tabs to display. Typically page developers place an instance of this portlet in the header of a portal shell to render a set of tabs that an end user can select.
Simple Property Editors Component	Property Editors. Page developers use this component to deploy a collection of legacy property editors. Page developers can use these editors when developing portlet views. Each editor provides a specialized user interface for editing a portlet preference on the portlet Properties page.

Name	Description
Spotlight Tab Styles Portlet	Page Components. Page developers indirectly use this portlet when using the Dynamic Tabs Portlet (wm_tabs) to display tabs as links with the select link displayed in the skin's "spotlight" style. For example, in some skins the "spotlight" style for a link might be outlined with an oval background.
Tab Styles Portlet	Template. Page designers use this portlet to dynamically create tabs that use a standard style.
Toggle Opened/Closed Portlet	Page components. End users use this portlet to toggle the visibility of an element in another portlet. The portlet is intended for use only with legacy portlets for which toggling capabilities are not available. It it not needed for newer portlets because the webMethods CAF toolkit provides several methods for toggling whether controls are visible or hidden.
Universal Picker Portlet	System. Portlet developers use this portlet to display a picker list of My webMethods Server items from which an end user can select one or more of the items.

Advanced Tooltip Portlet

Portlet Title	Advanced Tooltip
Portlet Name	wm_tooltip
Portlet File Name	wm_tooltip.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_tooltip
Default Instances of the portlet	None

UI Controls. Page developers use this portlet to provide an advanced tooltip for any HTML element in a page.

Properties

HTML For the Tooltip (tooltipMarkup)
Defines the text for the tooltip. Specify the HTML fragment to render as the body of the tooltip. If the property has no value, the portlet renders an empty tooltip.

Tooltip Background Color (bgcolor)

Defines the background color to use for the tooltip. Specify the HTML color code. The default is `#FFFFE1`, which is light yellow.

Target HTML Element ID (targetID)
Identifies the HTML element to which to attach the tooltip. Specify the ID of the HTML element. If the property has no value, the portlet does not attach the tooltip to any HTML element.

Calendar Picker Portlet

Portlet Title	Calendar Picker
Portlet Name	wm_calendarpicker
Portlet File Name	wm_calendarpicker.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.calendarpicker
Default Instances of the portlet	None

System. Page developers who are building 6.x proprietary, non-JSR168 portlets use this portlet to add the Date property editor to a page. The Date property editor renders a text input file with a calendar icon next to it. When an end user clicks either the input field or the icon, the Calendar Picker portlet displays a calendar popup window from which the end user can select a date. After selecting the date and closing the popup window, the portlet displays the selected date in the input field.

Properties

Form Name (form)
Identifies the name of the HTML form element that contains the Date property editor. Specify the name of the form. If the property has no value, the Date property editor will not function properly.

Property Name (property)
Required. Identifies the portlet property that holds the date for which you are providing the Date property editor. Specify the name of the portlet property.

Property Value (value)
Holds the current date value to initially display in the Date property editor. Specify the current date value either as a String or a long integer representing a Java date value. If you specify a String value, format the string with the pattern specified by the Date Format Pattern (pattern) property. If the property has no value, the Date property editor displays the value defined by the Default Value (defaultValue) property.

Read Only (readOnly)
Indicates whether an end user can use the Date property editor to modify the current value. Specify one of the following:

- `true` – The end user cannot modify the current value.
- `false` – Default. The end user can modify the current value.

CSS Class (style)
Identifies the style to apply to the text input that the Date property editor renders. Specify the CSS class. The default is `small`.

Date Format Pattern (pattern)
Defines the pattern to use to format the date. Specify a pattern that conforms to the pattern definition from the SimpleDateFormat Javadocs. The default is `yyyy-MM-dd HH:mm`.

Default Value (defaultValue)
Defines the value to initially display in the Date property editor when the Property Value (value) has no value. Specify a String value; the string does not have to match the pattern specified by the Date Format Pattern (pattern) property. For example, you might use `Please enter a date`. If this property and the Property Value (value) property have no value, the input text field of the Date property editor will be empty.

Allow Manual Entry (allowManual)
Indicates whether the end user can type a date in the input field or must use the calendar popup to select a date. Specify one of the following:

- `true` – Default. The end user can type a date in the input field.
- `false` – The end user cannot type a date in the input field; the end user must select the date using the calendar popup.

Dynamic Table Portlet

Portlet Title	Dynamic Table
Portlet Name	wm_table
Portlet File Name	wm_table.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.table
Default Instances of the portlet	None

System. Page developers using My webMethods Server version 6.x can use this portlet to display data in a table. For My webMethods Server version 7.x and later, use the Table controls that the webMethods CAF framework provides.

Page developers can embed the Dynamic Table portlet into a custom portlet using the `<portlet:portlet>` JSP tag, and then configure the datasource of the data to display in the table.

Alternatively, page developers can publish the Dynamic Table portlet as a standalone portlet and configure it to use My webMethods Server commands or wiring to provide the data to display.

If a page developer specifies more than one data property, the order of precedence is as listed in the following table:

Data Property	Precedence
command	Overrides all other properties.
sliceCommand	Overrides listCommand, slice, and list.
listCommand	Overrides slice and list.
slice	Overrides list.
list	No override applies.

Properties

List (list)
Identifies unsorted, unpaged data to display in the table. Specify the data using a list (java.util.List), array, comma-separated values (CSV) string, or XML Document Object Model (DOM) node.

To identify the data to display in the table, you should specify this List (list) property, the List Command (listCommand) property, the Slice (slice) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the table will be empty.

List Command (listCommand)
Identifies unsorted, unpaged data to display in the table. Specify a My webMethods Server command to invoke to produce the data to display. When you use this property, you can also specify:

- Resource (resource) property to identify a My webMethods Server resource against which to execute the command
- Parameters (parameters) property to specify properties for the command

To identify the data to display in the table, you should specify this List (list) property, the List Command (listCommand) property, the Slice (slice) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the table will be empty.

Slice (slice)
Identifies the sorted and paged data slice to display in the table. Specify the data using a list (java.util.List), array, comma-separated values (CSV) string, or XML Document Object Model (DOM) node.

To identify the data to display in the table, you should specify this Slice (slice) property, the List Command (listCommand) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the table will be empty.

Slice Command (sliceCommand)

Identifies sorted and paged data slice to display in the table. Specify a My webMethods Server command to invoke to produce the data to display. When you use this property, you can also specify:

- Resource (resource) property to identify a My webMethods Server resource against which to execute the command
- Parameters (parameters) property to specify properties for the command

To identify the data to display in the table, you should specify this Slice Command (sliceCommand) property, the List (list) property, the List Command (listCommand) property, or the Slice (slice) property. If none of these properties have values, the table will be empty.

Resource (resource)
Identifies the My webMethods Server resource against which to execute the command specified in the List Command (listCommand) or Slice Command (sliceCommand) property. Specify the thingID or alias of the resource.

If you specify the Portlet (portlet) property, it is used instead of this property. If neither this property nor the Portlet (portlet) property has a value, the List Command or Slice Command is invoked without a specific resource.

This property is ignored if you do not specify the List Command (listCommand) or Slice Command (sliceCommand) property.

Portlet (portlet)
Identifies the portlet to use for paging and sorting controls when the Dynamic Table portlet is used inside another legacy portlet. Specify the thingID or alias of the portlet. If the property has no value, it defaults to this portlet's thingID.

Parameters (parameters)
Identifies parameters to use with the command specified in the List Command (listCommand) or Slice Command (sliceCommand) property. Specify the parameters in the form of URL parameters, for example, *paramOne=valueOne&paramTwo=valueTwo*. If this property has no value, the command is executed with no additional parameters.

This property is ignored if you do not specify the List Command (listCommand) or Slice Command (sliceCommand) property.

Titles (titles)
Defines the titles, or headers, to use for the columns in the table. Specify a comma-separated list of titles. For example, for a table with four columns you might specify:

```
ID, Category, Description, Priority
```

If the property has no value, the table will not have column headers.

Keys (keys)
Defines the sort keys to use for each column of the table. A sort key is a property to use for sorting. Specify a comma-separated list of sort keys. If you do not specify a sort key for a column, that column is not sortable. For example, for a table with four columns you might specify the following where the third column is not sortable:

```
{ID}, {category}, , {priority}
```

If the property has no value, none of the columns will be sortable.

Values (values)
Defines the values to specify in each column. Specify a comma-separated list of the properties to display in each column. For example, for a table with four columns you might specify:

```
{ID}, {category}, {description}, {priority}
```

If the property has no value, the columns will be empty.

Links (links)
Defines how to link the data displayed in each column of the table. Specify a comma-separated list of links. If you do not specify a link for a column, the data in that column is displayed without a link. For example, for a table with four columns you might specify the following where only the first and second columns are linked:

```
http://example.com/items?id={ID}, http://example.com/categories?cat={category}, ,
```

If the property has no value, none of the column values will be linked.

Tooltips (tooltips>)
Defines the tooltips for each column's value. Specify a comma-separated list of text to use for tooltips. For example, for a table with four columns you might specify the following where only the second and third columns have tooltips:

```
, {categoryDescription}, , {priorityDescription}
```

If the property has no value, none of the column will have tooltips.

Alignments (alignments)
Defines the alignment to use for each column. Specify a comma-separated list of alignment values. For alignment values, you can specify the following `left`, `right`, `centered`, or specify nothing if you want to use the default value. The default alignment value is left. For example, for a table with four columns you might specify the following:

```
center, left, left, center
```

If the property has no value, all columns use default alignment value: left.

Widths (widths)
Defines the width of each column. Specify a comma-separated list of widths using either a percentage of the table or pixel value. For example, for a table with four columns you might specify:

```
5%, 30%, 50%, 15%
```

If the property has no value, column widths are determined by the client browser, based on the amount of content in each column.

Wrappings (wrappings)
Defines whether to wrap the data in each column. Specify a comma-separated list of wrap values. For wrap values:

- If you do not want the data to wrap specify, `nowrap`.

- If you want the data to wrap, specify nothing.

For example, for a table with four columns you might specify:

```
nowrap, nowrap, , nowrap
```

If the property has no value, all columns will wrap if necessary.

Selected (selected)
Defines how to select a row. Specify a row property to use for a selection key. For example, to select a row using its ID property, specify `{ID}`. If the property has no value, the rows of the table will not be selectable. A row is selected if its property value matches a value in the Selection (selection) property.

Selection (selection)
Identifies the values to use for selecting table rows. Specify a list of selection values. If the property has no value, no table rows are selected by default.

For example, if you specify `{ID}` for **Selected (selected)** and `1,3,5` for **Selection (selection)**, the rows with ID property value of 1, 3, or 5 are selected.

Allow Selection (allowSelection)
Indicates whether end users can select rows in the table. Specify one of the following:

- `true` – Users can select one row at a time.
- `false` – Default. Rows are not selectable.

Allow Multiple Selection (allowMultipleSelection)
Indicates whether end users can select multiple rows in the table at one time. Specify one of the following:

- `true` – Users can select multiple rows at one time.
- `false` – Default. Users cannot select multiple rows at one time.

The order of precedence between allowMultipleSelection and allowSelection is described in the following table:

	`allowMultipleSelection = false`	`allowMultipleSelection = true`
`allowSelection = false`	Users cannot select any rows.	Users can select multiple rows at a time.
`allowSelection = true`	Users can select one row at a time.	Users can select multiple rows at a time.

Start (start)
Defines the index of the first table row to display. Specify the index number, which is one-based; that is, specify `1` to display row one of the table. If the property has no value, it defaults to `1`.

Page Size (pageSize)

Defines the number of rows to display per page. Specify a whole number that is one or greater. If the property has no value, the portlet determines the number of items to display per page from the user preferences in the User Profile; the default user preference value is `10`.

Sort Key (sort)
Defines how to sort the table data. Specify the row property on which to sort. For example, to sort the table by the ID property, specify `{ID}`. If the property has no value, the table data is not sorted.

Sort Order (order)
Defines the order in which to sort the table data. Specify one of the following:

- `ascending` – Default. Sorts the table in ascending order.
- `descending` – Sorts the table data in descending order.

Query String (queryString)
Defines the query string that the table's datasource should use to generate the table data. The value you specify depends on the datasource. If the property has no value, a query string is not used. Note that most datasources do not require or use a query string.

View (view)
Defines the namespace to use for the paging cookie and/or paging links. Specify an arbitrary string to identify the namespace. If the property has no value, a special namespace is not used.

Style (style)
Defines the display style for the table. Specify one of the following:

- `standard` – Default. Displays the table in the standard My webMethods Server table style.
- `condensed` – Displays the table slightly condensed, depending on the skin that is used.
- `table` – Displays the table as a single HTML table.

Show Header (showHeader)
Indicates whether to display column headers. Specify one of the following:

- `true` – Default. Display column headers.
- `false` – Do not display column headers.

Show Footer (showFooter)
Indicates whether to display the paging footer for the table. Specify one of the following:

- `true` – Default. Display the paging footer.
- `false` – Do not display the paging footer.

Show Total (showTotal)
Indicates whether to display the total number of rows in the table. Specify one of the following:

- `true` – Default. Display the total number of rows.
- `false` – Do not display the total number of rows.

Show Total Selection (showTotalSelection)

Indicates whether to display the total number of selected rows in the table. Specify one of the following:

- `true` – Default. Display the total number of selected rows.
- `false` – Do not display the total number of selected rows.

Show Pages (showPages)
Indicates whether to display links to individual pages, (for example, ... 4 5 6 7 8 9 10 11 12 13 ...). Specify one of the following:

- `true` – Default. Display links to individual pages.
- `false` – Do not display links to individual pages.

Show Empty (showEmpty)
Indicates whether to display an empty message in the table contents when there is no table data to display. Specify one of the following:

- `true` – Default. Display an empty message when there is no table data to display.
- `false` – Display nothing in the table contents when there is no table data to display.

The default empty message is "Empty". If you want to display alternative text, use the emptyMessage property.

Properties that are Not Displayed in the User Interface

textTitles
Defines a set of alternate column headers to use when the table is exported. If you specify a value for this property, when the table is exported, the export uses the textTitles values in place of the column titles defined by the Titles (titles) property. Specify a comma-separated list of column titles, for example:

```
ID, Category, Description, Priority Number
```

You can use this property in conjunction with the textValues property to export a different set of columns than are normally displayed for the table.

If the property has no value, when the table is exported, the column titles defined by the Titles (titles) property are used.

textValues
Defines a set of alternate column values to use for a table export. If you specify a value for this property, when the table is exported, the values you define with the textValues property are exported rather than the values defined by the Values (values) property. Specify a comma-separated list of properties for example:

```
{ID}, {category}, {description}, {priority}
```

You can use this property in conjunction with the textTitles property to export a different set of columns than are normally displayed for the table.

If the property has no value, when the table is exported, the values defined by the Values (values) property are exported.

selectionPropertyName
Identifies alternate values to use for selecting table rows. Specify a list of selection values.

If you specify a value for this property, for example, "mySelectionProperty", the portlet uses the property with the specified name to determine the selected rows instead of the Selection (selection) property. If the property has no value, the Selection (selection) property is used.

onchange
Identifies client-side JavaScript to invoke when an end user selects or deselects a row. Specify the name of the JavaScript onchange handler. If the property has no value, custom JavaScript is not invoked when a table row selection changes.

cookie
Identifies a paging cookie to use instead of the values in the Start (start), Sort Key (sort), Sort Order (order), and Query String (queryString) properties. Specify the name of the paging cookie.

If the property has no value, a paging cookie is automatically generated and populated with the values specified in the Start (start), Sort Key (sort), Sort Order (order), and Query String (queryString) properties.

formPaging
Indicates whether to submit the form containing the table (with a void portlet-controller method) when an end user selects a paging link. Specify one of the following:

- `true` – Submit the form (with a void pca method) when an end user selects a paging link.
- `false` – Default. Use regular paging links when an end user selects a paging link.

emptyMessage
Defines the empty message. Specify the text to display in the table contents when there is no table data to display. The portlet uses the emptyMessage when the Show Empty (showEmpty) property is `true`. If the emptyMessage property has no value, the empty message defaults to `Empty`.

showExport
Indicates whether to display the **Export Table** button. Specify one of the following:

- `true` – Display the **Export Table** button.
- `false` – Do not display the **Export Table** button.

showPagingHeader
Indicates whether to show paging information in the table header or the table footer. Specify one of the following:

- `true` – Display the total number of rows, total selected rows, and the **Export Table** button in the table header.
- `false` – Default. Display the total number of rows, total selected rows, and the **Export Table** button in the table footer.

Note that the paging items displayed also depend on the values of the Show Total (showTotal), Show Total Selection (showTotalSelection), and showExport properties. The values of these properties must be `true` for the corresponding information to be displayed in the table header or table footer.

categorized
Indicates that you want to group rows into categories and defines how to categorize the rows. Specify the row property to use to categorize the table rows. All the rows that have the same value for the row property you specify will be in the same category. For example, to categorize rows by their priority property, specify `{priority}`. If the categorized property has no value, the portlet does not categorize rows.

Dynamic Tree Portlet

Portlet Title	Dynamic Tree
Portlet Name	wm_tree
Portlet File Name	wm_tree.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.tree
Default Instances of the portlet	None

System. Page developers using My webMethods Server version 6.x can use this portlet to display data in a tree. For My webMethods Server version 7.x and later, use the Tree controls that the webMethods CAF framework provides.

Page developers can embed the Dynamic Tree portlet into a custom portlet using the `<portlet:portlet>` JSP tag, and then configure the datasource of the data to display in the tree.

Alternatively, page developers can publish the Dynamic Tree portlet as a standalone portlet and configure it to use My webMethods Server commands or wiring to provide the data to display.

If a page developer specifies more than one data property, the order of precedence is described in the following table:

Data Property	Precedence
command	Overrides all other properties.
sliceCommand	Overrides listCommand, slice, and list.
listCommand	Overrides slice and list.
slice	Overrides list.

Data Property	Precedence
list	No override applies.

Properties

List (list)
Identifies unsorted data to display in the tree. Specify the data using a list (java.util.List), array, comma-separated values (CSV) string, or XML Document Object Model (DOM) node.

To identify the data to display in the tree, you should specify this List (list) property, the List Command (listCommand) property, the Slice (slice) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the tree will be empty.

List Command (listCommand)
Identifies unsorted data to display in the tree. Specify a My webMethods Server command to invoke to produce the data to display. When you use this property, you can also specify:

- Resource (resource) property to identify a My webMethods Server resource against which to execute the command
- Parameters (parameters) property to specify properties for the command

To identify the data to display in the tree, you should specify this List Command (listCommand) property, the List (list) property, the Slice (slice) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the tree will be empty.

Slice (slice)
Identifies the sorted data slice to display in the tree. Specify the data using a list (java.util.List), array, comma-separated values (CSV) string, or XML Document Object Model (DOM) node.

To identify the data to display in the tree, you should specify this u property, the List (list) property, the Slice (slice) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the tree will be empty.

Slice Command (sliceCommand)
Identifies the sorted data slice to display in the tree. Specify a My webMethods Server command to invoke to produce the data to display. When you use this property, you can also specify:

- Resource (resource) property to identify a My webMethods Server resource against which to execute the command
- Parameters (parameters) property to specify properties for the command

To identify the data to display in the tree, you should specify this Slice Command (sliceCommand) property, the List (list) property, the List Command (listCommand) property, or the Slice (slice) property. If none of these properties have values, the tree will be empty.

Resource (resource)
Identifies the My webMethods Server resource against which to execute the command specified in the List Command (listCommand) or Slice Command (sliceCommand) property. Specify the thingID or alias of the resource.

If you specify the Portlet (portlet) property, it is used instead of this property. If neither this property nor the Portlet (portlet) property has a value, the List Command or Slice Command is invoked without a specific resource.

This property is ignored if you do not specify the List Command (listCommand) or Slice Command (sliceCommand) property.

Portlet (portlet)
Identifies the portlet to use for paging and sorting controls when the Dynamic Tree portlet is used inside another legacy portlet. Specify the thingID or alias of the portlet. If the property has no value, it defaults to this portlet's thingID.

Parameters (parameters)
Identifies parameters to use with the command specified in the List Command (listCommand) or Slice Command (sliceCommand) property. Specify the parameters in the form of URL parameters, for example, *paramOne=valueOne&paramTwo=valueTwo*. If this property has no value, the command is executed with no additional parameters.

This property is ignored if you do not specify the List Command (listCommand) or Slice Command (sliceCommand) property.

Titles (titles)
Defines the column headers to use for the columns in the tree. Specify a comma-separated list of titles. For example, for a tree with four columns you might specify:

```
ID, Category, Description, Priority
```

If the property has no value, the tree will not have column headers.

Keys (keys)
Defines the sort keys to use for each column of the tree. A sort key is the property to use to sort the column. Specify a comma-separated list of sort keys. If you do not specify a sort key for a column, that column is not sortable. For example, for a tree with four columns you might specify the following where the third column is not sortable:

```
{ID}, {category}, , {priority}
```

If the property has no value, none of the columns will be sortable.

Values (values)
Defines the values to specify in each column. Specify a comma-separated list of the properties to display in each column. For example, for a tree with four columns you might specify:

```
{ID}, {category}, {description}, {priority}
```

If the property has no value, the columns will be empty.

Links (links)
Defines how to link the data displayed in each column of the tree. Specify a comma-separated list of links. If you do not specify a link for a column, the data in that column is displayed without a link. For example, for a tree with four columns you might specify the following where only the first and second columns are linked:

```
http://example.com/items?id={ID}, http://example.com/categories?cat={category}, ,
```

If the property has no value, none of the column values will be linked.

Tooltips (tooltips)
Defines the tooltips for each column's value. Specify a comma-separated list of text to use for tooltips. For example, for a tree with four columns you might specify the following where only the second and third columns have tooltips:

```
, {categoryDescription}, , {priorityDescription}
```

If the property has no value, none of the column will have tooltips.

Alignments (alignments)
Defines the alignment to use for each column. Specify a comma-separated list of alignment values. For alignment values, you can specify the following: `left`, `right`, `centered`, or specify nothing if you want to use the default value. The default alignment value is left. For example, for a tree with four columns you might specify the following:

```
center, left, left, center
```

If the property has no value, the columns use default alignment value: left.

Widths (widths)
Defines the width of each column. Specify a comma-separated list of widths using either a percentage of the table or pixel value. For example, for a tree with four columns you might specify:

```
5%, 30%, 50%, 15%
```

If the property has no value, column widths are determined by the client browser, based on the amount of content in each column.

Wrappings (wrappings)
Defines whether to wrap the data in each column. Specify a comma-separated list of wrap values. For wrap values:

- If you do not want the data to wrap specify, `nowrap`.
- If you want the data to wrap, specify nothing.

For example, for a tree with four columns you might specify:

```
nowrap, nowrap, , nowrap
```

If the property has no value, all columns will wrap if necessary.

Row Identifier (rowID)
Identifies the unique identifier to use for a row. Specify a row property. For example, to use a row's ID property as the unique identifier, specify `{ID}`. If the property has no value, the tree is displayed as a flat list. This property is required, along with the Child References (childIDRefs) and Parent Reference (parentIDRef) properties to establish the tree hierarchy.

Child References (childIDRefs)

Defines how to establish child rows. Specify a row property that lists the IDs of the children for a row. The list must be either a comma-separated or whitespace-separated list of row IDs. For example to use a row's childIds property, specify `{childIds}`.

One of the Child References (childIDRefs), Parent Reference (parentIDRef), Children (children), or Parent (parent) properties must be specified in order to establish the tree hierarchy. Otherwise, the tree will be displayed as a flat list of rows.

Parent Reference (parentIDRef)
Defines how to establish parent rows. Specify a row property that contains the ID of the row's parent. For example, to use a row's parentId property, specify `{parentId}`.

One of the Child References (childIDRefs), Parent Reference (parentIDRef), Children (children), or Parent (parent) properties must be specified in order to establish the tree hierarchy. Otherwise, the tree will be displayed as a flat list of rows.

Children (children)
Defines the children for a row. Specify a row property that contains the children of the row (as a list, array, or XML Document Object Model (DOM) node list of child rows). For example, to use a row's children property, specify `{children}`.

One of the Child References (childIDRefs), Parent Reference (parentIDRef), Children (children), or Parent (parent) properties must be specified in order to establish the tree hierarchy. Otherwise, the tree will be displayed as a flat list of rows.

Parent (parent)
Defines the parent for a row. Specify a row property that contains the parent of the row. For example, to use a row's parent property, specify `{parent}`.

One of the Child References (childIDRefs), Parent Reference (parentIDRef), Children (children), or Parent (parent) properties must be specified in order to establish the tree hierarchy. Otherwise, the tree will be displayed as a flat list of rows.

Selected (selected)
Defines how to select a row. Specify a row property to use for a selection key. For example, to select a row using its ID property, specify `{ID}`. If the property has no value, the rows of the tree will not be selectable. A row is selected if its property value matches the value in the Selection (selection) property

Selection (selection)
Identifies the values to use for selecting tree rows. Specify a list of selection values. If the property has no value, no table rows are selected by default.

For example, if you specify `{ID}` for **Selected (selected)** and `1,3,5` for **Selection (selection)**, the rows with ID property value of 1, 3, or 5 are selected.

Allow Selection (allowSelection)
Indicates whether end users can select multiple rows in the tree at one time. Specify one of the following:

- `true` – Users can select multiple rows at a time.
- `false` – Default. Rows are not selectable.

Store Opened State (storeOpenedState)
Indicates whether to persist the open state when an end user re-sorts or submits the form containing the tree. Specify one of the following:

- `true` – Persist the open state.
- `false` – Default. Do not persist the open state.

Is Open (open)
Defines how to determine whether to consider a row as open. Specify a row property. How the portlet uses the row property you specify depends on the value of the Store Opened State (storeOpenedState) property:

- When Store Opened State (storeOpenedState) is `true` indicating to persist the row open state:

  A row is considered open if the value of the row property you specify for Is Open (open) matches a value in the Opened (opened) property. For example, to use the ID property by specifying `{ID}` for this property, a row is open if its ID property matches a value that you specify for the Opened (opened) property.

  If the Opened (opened) property has no value, the opened state will not be stored.

- When Store Opened State (storeOpenedState) is `false` indicating to *not* persist the row open state:

  A row is considered open if the value of the property you specify for Is Open (open) evaluates to `true`. For example, to use the open property by specifying `{open}`, a row is open if the value of its open property is `true`.

  If the Opened (opened) property has no value, no rows will be displayed initially as open.

Opened (opened)
Defines the rows to open by default when the Store Opened State (storeOpenedState) property is `true`. This property is ignored when the Store Opened State (storeOpenedState) property is `false`.

Specify a list of opened row keys. For example, if you specify `{ID}` for **Is Open (open)** and `1,3,5` for **Opened (opened)**, the rows with ID property value of 1, 3, or 5 are initially displayed as opened.

If the property has no value, rows will only be opened if the Store Opened State (storeOpenedState) property is `true` and there are open row in the saved state.

Sort Key (sort)
Defines how to sort the tree data. Specify the row property on which to sort. For example, to sort the tree by the ID property, specify `{ID}`. If the property has no value, the tree data is not sorted.

Sort Order (order)
Defines the order in which to sort the tree data. Specify one of the following:

- `ascending` – Default. Sorts the tree in ascending order.
- `descending` – Sorts the tree data in descending order.

Query String (queryString)
Defines the query string that the tree's datasource should use to generate the tree data. The value you specify depends on the datasource. If the property has no value, a query string is not used. Note that most datasources do not require or use a query string.

View (view)
Defines the namespace to use for the paging cookie and/or paging links. Specify an arbitrary string to identify the namespace. If the property has no value, a special namespace is not used.

Style (style)
Defines the display style for the tree. Specify one of the following:

- `standard` – Default. Displays the table in the standard My webMethods Server table style.
- `condensed` – Displays the table slightly condensed, depending on the skin that is used.
- `table` – Displays the table as a single HTML table.

Show Header (showHeader)
Indicates whether to display column headers. Specify one of the following:

- `true` – Default. Display column headers.
- `false` – Do not display column headers.

Show Total Selection (showTotalSelection)
Indicates whether to display the total number of selected rows in the tree. Specify one of the following:

- `true` – Display the total number of selected rows.
- `false` – Default. Do not display the total number of selected rows.

Show Empty (showEmpty)
Indicates whether to display an empty message in the tree contents when there is no tree data to display. Specify one of the following:

- `true` – Default. Display an empty message when there is no tree data to display.
- `false` – Display nothing in the tree contents when there is no tree data to display.

The default empty message is "Empty". If you want to display alternative text, use the emptyMessage property.

Controls Column (controlsColumn)
Identifies the column of the tree in which to display open and close controls. Specify the zero-based index number of the column. The default is `0`, indicating to display the open and close controls in the first column.

Properties that are Not Displayed in the User Interface

selectionPropertyName
Identifies alternate values to use for selecting tree rows. Specify a list of selection values.

If you specify a value for this property, for example, "mySelectionProperty", the portlet uses the property with the specified name to determine the selected rows instead of the Selection (selection) property. If the property has no value, the Selection (selection) property is used.

openedPropertyName
Identifies the row property to use to consider a row open instead of the value of the Opened (opened) property. Specify the name of the property. If the property has no value, the Opened (opened) property is used.

cookie
Identifies a paging cookie to use instead of the values in the Sort Key (sort), Sort Order (order), and Query String (queryString) properties. Specify the name of the paging cookie.

If the property has no value, a paging cookie is automatically generated and populated with the values specified in the Sort Key (sort), Sort Order (order), and Query String (queryString) properties.

emptyMessage
Defines the empty message. Specify the text to display in the tree contents when there is no table data to display. The portlet uses the emptyMessage when the Show Empty (showEmpty) property is `true`. If the emptyMessage property has no value, the empty message defaults to `Empty`.

Dynamic Tabs Portlet

Portlet Title	Dynamic Tabs
Portlet Name	wm_tabs
Portlet File Name	wm_tabs.pdp
Top-level Folder	template
JSR168 Portlet?	No
Alias	portlet.tabs
Default Instances of the portlet	None

System. Page developers using My webMethods Server version 6.x can use this portlet to display a configured list of tabs. For My webMethods Server version 7.x and later, use the Tab controls that the webMethods CAF framework provides.

Page developers can embed the Dynamic Tabs portlet into a custom portlet using the `<portlet:portlet>` JSP tag, or alternatively, publish the Dynamic Table portlet as a standalone portlet and configure it to use My webMethods Server commands or wiring to provide the data to display.

If a page developer specifies more than one data property, the order of precedence is as follows:

Data Property	Precedence
sliceCommand	Overrides listCommand, slice, and list.

Data Property	Precedence
listCommand	Overrides slice and list.
slice	Overrides list.
list	No override applies.

Properties

Slice (slice)
Identifies the data to use for the tab. Specify sorted list slice data.

To identify the data to use for the tabs, you should specify this Slice (slice) property, the List (list) property, the Slice Command (sliceCommand) property, or the List Command (listCommand). property. If none of these properties have values, the tabs will be empty.

List (list)
Identifies the data to use for the tab. Specify unsorted, unpaged list data.

To identify the data to use for the tabs, you should specify this List (list) property, the Slice Command (sliceCommand) property, or the List Command (listCommand). property. If none of these properties have values, the tabs will be empty.

Slice Command (sliceCommand)
Identifies the data to use for the tab. Specify a My webMethods Server command to invoke to produce the sorted list slice data to use for the tabs.

To identify the data to use for the tabs, you should specify this Slice Command (sliceCommand) property, the Slice (slice) property, the List (list) property, or the List Command (listCommand) property. If none of these properties have values, the tabs will be empty.

List Command (listCommand)
Identifies the data to use for the tab. Specify a My webMethods Server command to invoke to produce the unsorted, unpaged list data to use for the tabs.

To identify the data to use for the tabs, you should specify this List Command (listCommand) property, the Slice (slice) property, the List (list) property, or the Slice Command (sliceCommand) property. If none of these properties have values, the tabs will be empty.

Parameters (parameters)
Identifies parameters to use with the command specified in the List Command (listCommand) or Slice Command (sliceCommand) property. Specify the parameters in the form of URL parameters, for example, paramOne=valueOne&paramTwo=valueTwo. If this property has no value, the command is executed with no additional parameters. This property is ignored if you do not specify the List Command (listComand) or Slice Command (sliceCommand) property.

Titles (titles)
Defines the titles to use for each tab. Specify a comma-separated list of tab titles. If the property has no value, the tabs will be displayed as empty.

Values (values)

Defines the IDs to assign to each tab. You can use the IDs to reference tab information, for example, the Title of a tab. Specify a comma-separated list of IDs. If the property has no value, IDs will not be assigned to the tabs.

Tooltips (tooltips)
Defines the tooltips to display for each tab. Specify a comma-separated list of tooltips. If the property has no value, the tabs will not have tooltips.

Links (links)
Defines links to use for each tab. Specify a comma-separated list of links. If the property has no value, the tabs will not be linked.

Selection Value (selection)
Identifies the values to use for selecting tabs. Specify a list of selection values. If the property has no value, no tabs are selected by default.

For example, if the Selection Value (selection) property is equal to "`1,3,5`", tabs whose Values (values) property is 1, 3, or 5 are selected.

Link Selected Tab (selectionLinked)
Indicates whether to enable the link for the selected tab. Specify one of the following:

- `true` – Default. Enable the link for the selected tab.
- `false` – Do not enable the link for the selected tab.

Style (style)
Defines the display style for the tab. Specify one of the options, listed in the following table:

Option	Description
`Tabs (portlet.tabstyles)`	Default. Displays standard tabs.
`Subtabs (portlet.tabstyles?style=condensed)`	Displays condensed tabs.
`Horizontal (portlet.tabstyles?layout=horizontal)`	Displays a horizontal list of links.
`Vertical (portlet.tabstyles?layout=vertical)`	Displays a vertical list of links.
`Horizontal - Bars (portlet.tabstyles? layout=horizontal& separator=|)`	Displays a horizontal list of links, with each link separated by a vertical bar.
`Popup (portlet.tabstyles.popup? text=[selected] defaultText=[name])`	Displays a single link with the selected item. When clicked, displays a pop-up menu to the side of the portlet showing all the links.
`Dropdown (portlet.tabstyles.popup? text=[name]&`	Displays a single link with the name of the portlet instance. When clicked, displays a

Option	Description
`dropdown=true)`	drop-down menu below the portlet with all the links.
`Hover (portlet.tabstyles.popup? text=[name]& layout=hover)`	Displays a single link with the name of the portlet instance. When hovered over, displays a drop-down menu below the portlet with all the links.

Start (start)
Defines the index of the first tab to display on the page. Specify the one-based index number; that is, specify `1` to display the first tab. If the property has no value, it defaults to `1`.

Page Size (pageSize)
Defines the number of tabs to display per page. Specify a whole number that is one or greater. If the property has no value, the portlet determines the number of items to display per page from the user preferences in the User Profile. The default user preference value is 10.

Total (total)
Defines the total number of tabs. This total can be calculated automatically if the tabs are specified by using the List (list) or List Command (listCommand) properties. However, they cannot be calculated automatically by using the Slice (slice) or Slice Command (sliceCommand) properties.

If the Slice (slice) or Slice Command (sliceCommand) property is used and the Total (total) property is not specified, then the total number of tabs is unknown and a Next link is always displayed, even if there are no more tabs to display.

Sort Key (sort)
Defines how to sort the tabs. Specify one of the properties you specified for the Values (values) property for the sort key. If the property has no value, the tabs are not sorted.

Sort Order (order)
Defines the order in which to sort the tabs. Specify one of the following:

- `ascending` – Default. Sorts the tabs in ascending order.
- `descending` – Sorts the tabs in descending order.

Query String (queryString>)
Defines the query string that the tab's datasource can use to generate the list of tabs. The value you specify depends on the datasource. If the property has no value, a query string is not used. Note that most datasources do not require or use a query string.

View Name (view)
Defines the namespace to use for the sorting and paging properties. Specify an arbitrary string to identify the namespace. The value you specify must be unique within the portlet that is rendering the tabs. If the property has no value, the default namespace (wm_tabs) will be used.

Resource (resource)

Identifies the My webMethods Server resource against which to execute the command specified in the List Command (listCommand) or Slice Command (sliceCommand) property. Specify the thinkID or alias of the resource.

If you specify the Portlet (portlet) property, then the Portlet (portlet) property is used instead of the Resource (resource) property.

If neither the Resource (resource) nor the Portlet (portlet) property has a value, then the List Command (listCommand) or Slice Command (sliceCommand) property is invoked without a specific resource.

The Resource (resource) property is ignored if you do not specify the List Command (listCommand) or Slice Command (sliceCommand) property.

Portlet (portlet)
Identifies the portlet to use for paging and sorting controls when the Dynamic Tabs portlet is used inside another legacy portlet. Specify the thingID of the portlet. This is needed to render the paging and sorting controls. If the property has no value, it defaults to this portlet's thingID.

Extended Property Editors Component

Portlet Title	Extended Property Editors
Portlet Name	wm_extendedpropertyeditors
Portlet File Name	wm_extendedpropertyeditors.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

Property Editors. Page developers use this component to provide specialized user interfaces for editing portlet preferences on the Portlet Properties page. This component is a collection of custom editors that other portlets can reuse.

Portlet Property Editors Provided with the Extended Property Editors Component

The following table lists the collection of legacy portlet property editors provided with the Extended Property Editors component.

Editor	Description
directory	Provides a combobox from which an end user can select a directory service.
dsn	Provides a combobox from which an end user can select a JDBC datasource.
file-encoding	Provides a combobox from which an end user can select a file encoding.

Editor	Description
file-size	Provides formatted text that displays a file size.
image_source	Provides a panel for specifying the source of an image file.
picker_pca	Provides a combobox from which an end user can select a legacy PCA target.
renderer	Provides a combobox from which an end user can select the renderer to use to display the resource.
shell	Provides a combobox from which an end user can select a My webMethods Server shell.
skin	Provides a combobox from which an end user can select a My webMethods Server skin.

I18N Property Editors Component

Portlet Title	I18N Property Editors
Portlet Name	wm_i18npropertyeditors
Portlet File Name	wm_i18npropertyeditors.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.i18npropertyeditors
Default Instances of the portlet	None

Property Editors. Page developers use this component to provide specialized user interfaces for editing internationalization-specific portlet preferences on the Portlet Properties page. This component is a collection of custom editors that other portlets can reuse.

The following table lists the collection of legacy property editors provided with the I18N Property Editors Component.

Legacy Property Editor	Description
Country/Region ID Chooser	Provides a drop-down list from which an end user can select a country or region (for example, "Argentina" or "Antarctica").
Locale Chooser	Provides a drop-down list from which an end user can select a locale (for example, "es-AR Spanish (Argentina)" or "en-US English (United States)").

Legacy Property Editor	Description
Time Zone Chooser	Provides a drop-down list from which an end user can select a time zone (for example, "America/Argentina/Buenos Aries (Argentine Time)" or "America/Chicago (Central Standard Time)").

Page developers configure this component to add a drop-down list for one of the internationalization properties. For example, to provide an editor for the Country/Region ID, the page developer configures the component to use the type `Country/Region ID Chooser`. As a result, the view will contain a drop-down list that contains values a user can select for the Country/Region ID property.

For a sample of using this component, see the source for wm_xt_userprofileprovider, which is located in the components\services\directory\services directory. The following is a sample XML from its wm-portlet.xml file:

```
<preference>
    <name>country_region_id</name>
    <display-name>Country/Region ID</display-name>
    <group>attributes</group>
    <scope>content</scope>
    <property-editor type="Country/Region ID Chooser">
        <param>
            <name>validationRegexpMsg</name>
            <value />
        </param>
        <param>
            <name>validationRegexp</name>
            <value />
        </param>
        <param>
            <name>dhtml</name>
            <value>true</value>
        </param>
        <param>
            <name>style</name>
            <value>medium</value>
        </param>
        <param>
         <name>onchange</name>
            <value />
        </param>
    </property-editor>
    <hidden>false</hidden>
    <required>false</required>
    <dbo-type>
        <data-type>0</data-type>
        <data-length>3</data-length>
    </dbo-type>
</preference>
```

Properties for the Editor

type (type)

Required. Identifies the property editor type, which indicates the type of list to provide. Specify `Country/Region ID Chooser`, `Locale Chooser`, or `Time Zone Choose`.

style (style)
Required. Defines the text size for items in the list. Specify small, `medium`, or large. The default is `medium`.

onchange (onchange)
Defines the action to take when an end user updates the drop-down list. Specify a command that you defined in the portlet code. The default is blank, which indicates that no action is taken.

Legacy Property Editors Component

Portlet Title	Legacy Property Editors
Portlet Name	wm_legacypropertyeditors
Portlet File Name	wm_legacypropertyeditors.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.legacypropertyeditors
Default Instances of the portlet	None

Property Editors. Portlet developers use this component to provide specialized user interfaces for editing portlet preferences on the Portlet Properties page. This component is a collection of legacy property editors that are provided for backwards compatibility with earlier releases of My webMethods Server. They map old property editors to the new property editors that provide the equivalent functionality.

The Machine Skin

Component Title	The Machine
Component Name	wm_skin_machine
Component File Name	wm_skin_machine.skin
Top-level Folder	system

Skins. This component is a skin that is installed, but not used by default. Page developers can use this skin as the default skin or the target of a Skin Rule. Additionally, page developers can use the skin as the base for a custom skin.

The Machine component is a set of CSS style sheets and graphics that My webMethods Server can use when rendering a page.

Note:
The Machine skin was the default skin that My webMethods Server version 6.x used.

PCA Submit Button Portlet

Portlet Title	PCA submit button
Portlet Name	wm_pca_propertyeditors
Portlet File Name	wm_pca_propertyeditors.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	/portlet/wm_pca_propertyeditors
Default Instances of the portlet	None

Property Editors. Page developers use this component to provide specialized legacy Portlet-Controller Architecture (PCA) controls. This component is a collection of custom editors that other legacy portlets can reuse.

The following table lists the collection of legacy property editors provided with the I18N Property Editors Component:

Legacy Property Editor	Description
pcasubmit	Submit button for a PCA Form.
pcaregisterinput	Registers a non-property-editor input with the same name within a PCA Form.

Popup Tab Styles Portlet

Portlet Title	Popup Tab Styles
Portlet Name	wm_tabstyles_popup
Portlet File Name	wm_tabstyles_popup.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.tabstyles.popup
Default Instances of the portlet	None

Page Components. Page developers indirectly use this portlet when using the Dynamic Tabs Portlet (wm_tabs) to display popup style tabs. The Popup Tab Styles Portlet displays target text, and as the end user clicks or hovers over the target text, the portlet displays its list of tabs as a popup menu.

Properties

Layout (layout)
Indicates whether to display the menu when the user clicks the target text or when the user hovers over the target text. Specify one of the following:

- `popup` – Default. Display the menu when the user clicks the target text.
- `hover` – Display the menu when the user hovers over the target text.

Properties that are not Displayed in the User Interface

defaultText
Defines the target text to display when there is *no* selected tab; that is, when Selected (selected) is `-1`. Specify one of the following:

- *string* – The portlet displays the literal string you specify.
- `[name]` – The portlet displays the name of the Tabs (wm_tabs) portlet instance.

The default is a blank string.

text
Defines text to display when there is a selected tab; that is, when Selected (selected) is *not* `-1`. Specify one of the following:

- *string* – The portlet displays the literal string you specify.
- `[selected]` – The portlet displays the title of the selected tab.
- `[name]` – The portlet displays the name of the Tabs (wm_tabs) portlet instance.

The default is a blank string.

dropdown
Indicates whether to display the popup menu as a drop-down list. Specify one of the following:

- `true` – Display the popup menu as a drop-down list; that is, display it below the target text.
- `false` – Default. Display the popup menu to the right of the target text.

icon
Identifies the icon to display next to the target text. Specify the URL of the icon. The default is to display an arrow icon.

Properties Configured by the Tabs Portlet

The Tabs (wm_tabs) portlet automatically configures the following properties to provide the raw list of tabs to display in a popup style.

Titles (titles)
Defines the list of titles to use for the tabs. If the property has no value, no tabs are displayed when the user clicks on or hovers over the target text.

Tooltips (tooltips)
Defines the list of tool tips for the tabs. If the property has no value, no tool tips are displayed.

Links (links)
Defines the list of links to use for the tabs. If the property has no value, the tabs will not have links.

Next (next)
Identifies the "Next" link used to display the next page of tabs. If the property has no value, a "Next" link is not displayed.

Previous (prev)
Identifies the "Previous" link used to display the previous page of tabs. If the property has no value, a "Previous" link is not displayed.

Selected (selected)
Identifies the index of the tab that is to be selected when rendering the tabs. When **Selected** is `-1`, no tab is selected. The default is `-1`.

Progress Indicator Portlet

Portlet Title	Progress Indicator
Portlet Name	wm_progressindicator
Portlet File Name	wm_progressindicator.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

UI Controls. This is a deprecated portlet. The portlet displays a user interface control that shows a progress animation. The portlet was used as part of the legacy common search framework. Some existing portlets that have not been ported to webMethods CAF still use this portlet.

UI Options Properties

Context ID (contextID)
Uniquely identifies the progress indicator element on the client side. Specify an ID. If there are multiple progress indicator elements on the same page, each instance must have a different value for this property. If the property has no value, "id1" is used for the context ID.

Resource List View Portlet

Portlet Title	Resource List View
Portlet Name	wm_listview
Portlet File Name	wm_listview.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.listview
Default Instances of the portlet	None

Template. Page developers use this portlet in a shell section, portal page, or legacy portlet to display a list of items in a table format. The portlet retrieves the My webMethods Server items to display in the table by executing a My webMethods Server command. The page developer configures the columns to display in the table.

My webMethods Server uses this Resource List View portlet to draw the user interface for some of the legacy portlets. However, in JSR168 portlets, rather than using the Resource List View portlet, a page developer should use a Table control that is bound to a com.webmethods.caf.faces.data.portal.PortalContainerModel (or other appropriate content-provider).

Properties

Cache Content (cacheAge)
Defines how long to cache the contents of the portlet. Specify one of the following:

- `-1` – Default. Indicates that you want to cache the contents of the portlet indefinitely.
- `0` – Indicates that you do not want to cache the contents of the portlet.
- *minutes* – Defines the number of minutes to cache the contents of the portlet.

All instances of the portlet share the value of this property.

Command (command)
Identifies how to retrieve the items to display. Specify the command the portlet is to execute to retrieve the list of items. If the property has no value, the portlet displays an empty list.

Column Configuration (columns)

Describes the column configuration for the table that displays the retrieved data. Specify a comma-separated values (CSV) string that uses semicolons (;) to separate the columns and commas to separate the properties of each column. The following table lists the properties that you must specify for each column:

Property 1	Data to display	Specify one of the following: *name* Specify the name of the property that holds the data to display in the column. For example, if you want to display an item's name in the column, specify `name`. If you want to display an item's last modified date, specify `lastModifiedDate`. `icon` Indicates that you want the column to contain the item's icon. `menu` Indicates that you want the column to contain the item's thing menu.
Property 2	Column title	Specify the title to use for the column header, for example, `Name`.
Property 3	Column width	Specify the percentage of the table to use for the column, for example, `30%`.
Property 4	Horizontal Alignment	Specify how to horizontally align the data in the column. You can specify `left`, `center`, or `right`.
Property 5	Link	Specify what to use to link the contents of the column. ■ Use `{uri}` to indicate the item's standard ID should be used as the full link. ■ Define the link by using `{resourceURI}` and `{servletPath}` tokens. ■ `{resourceURI}` indicates to dynamically use the ID of the current item ■ `{servletPath}` indicates to dynamically use the ID of the servlet path For example, `{servletPath}{resourceURI}?layout=properties` indicates to link to the item's standard properties page.
Property 6	Sortable	Indicate whether a user can sort the table based on the contents of the column. Specify either `true` or `false`.

If the property has no value, the portlet displays an empty list.

The following is an example of a CSV string that provides the full configuration for a table with three columns, one that displays an item's name, one to display an item's description, and a third to display the item's last modified date.

```
name,Name,30%,left,{uri},true,;description,Description,40%,left,,true,;
lastModifiedDate,Last Modified,30%,right,,true,;
```

Visual Style (visualStyle)
Defines the list style. Specify one of the following:

- `1` – Default. Use a standard list.
- `2` – Use a condensed list, which might be the same as a standard list, depending on the current skin.

Page Limit (pageSize)
Defines the maximum number of items to display per page. Specify one of the following:

- `0` – Default. Use the current user's preferred number of items from the User Profile.
- *number* – The number of items to display per page. Specify a whole number greater than zero.

Portal Resource (listViewTargetURI)
Identifies the resource to which to apply the command that retrieves the items to display. Specify a My webMethods Server resource. If the property has no value, the portlet executes the command against the current page.

View Name (view)
Defines the view name to use for the paging cookie. Specify a view name. If the property has no value, the view name is blank; that is, there will be no view name.

Menu Configuration (menus)
Describes the custom menu configuration, which allows you to add custom menu items to the standard thing menus on a per-xtype basis. Specify a comma-separated values (CSV) string that uses semicolons (;) to separate menu definitions and commas to separate the properties of each menu. Th following table lists the properties that you must specify for each menu:

Property 1	xtype	Specify the xtype for which the menu definition applies, for example, `folder` or `wm_xt_fabricfolder`.
Property 2	items	Specify the items you want to add to the menu. For each menu item you want to add, provide a *title=link* pair. To add multiple menu items, separate each using an ampersand (&). The following is an example: `Add to Tools=javascript:myAddToTools()&Cast Magic Missile=javascript:myCastMagicMissile()`
Property 3	properties	Specify properties that you want to add to the menu item's thing object. For each property you want to add, provide a *javascriptProperty=thingProperty* pair. To add multiple properties,

separate each using an ampersand (&). The following is an example:

```
myDepth=depth&myInfo=info
```

If the property has no value, items will not have custom menus.

Custom Parameters (parameters)
Defines the parameters to use for the command that the portlet executes to get the items to display in the table. Note that parameters are command-specific. Specify the parameters like URL parameters, for example:

```
includeItems=true&extraInfo=true
```

If the property has no value, the portlet uses no extra parameters for the command.

Selected List View Portlet

Portlet Title	Selected List View
Portlet Name	wm_selectedlistview
Portlet File Name	wm_selectedlistview.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.selectedlistview
Default Instances of the portlet	None

Template. Page developers use this portlet to display tabs that represent the folders in a hierarchy. The page developer identifies the folders for which to display tabs by specifying the root folder of the hierarchy. At run time, the portlet dynamically determines the list of tabs to display. Typically page developers place an instance of this portlet in the header of a portal shell to render a set of tabs that an end user can select.

Properties

Root Folder (root)
Required. Identifies the folder containing the items for which to render tabs. Specify the thingID or alias of the folder. If the property has no value, the portlet displays an error message.

Leaf Item (target)
Identifies the item that the portlet is to render as the selected tab. Specify the thingID or alias of the item. The default is `current.resource`.

Depth (depth)

Indicates how deep in the folder structure the items for which you want to render tabs are in relation to the specified root folder. For example, if the items for which you want to render tabs are grandchildren of the root folder, set the depth to `2`. The default is `0`.

Titles (titles)
The expressions for this portlet are the same as those for the wm_table portlet, for example, (property-name), where "property-name" is the name of a property for a displayed item, like "name" or "description".

Indicates what to use for the text on each rendered tab. Specify an expression that resolves to the text. The default is `{name}`.

Tooltips (tooltips)
Indicates the tool tip text to display for each rendered tab. Specify an expression that resolves to the text. The default is `{tooltip}`.

Links (links)
Indicates the link to use for each rendered tab. Specify an expression that resolves to the link. The default it `{$link}`.

Style (style)
Indicates the CSS style to use for each rendered tab. Specify an expression that resolves to the CSS style. If the property has no value, no style is applied.

Start (start)
Defines the first tab to display when the number of tabs is larger than the value of the Page Size (pageSize) property, making it is necessary to page forward and back through the tabs. Specify the start index of the first tab. If the property has no value, the portlet behaves as if **Start** is set to `0`.

Page Size (pageSize)
Defines the number of tabs to display. Specify a positive whole number. If the number of tabs to display exceeds **Page Size**, the portlet provides links to page forward and backward through the set of tabs. If the property has no value, the portlet use the **Items Per Page** value from the current user's User Profile.

Sort Key (sort)
Indicates the key to use to sort the tabs. Specify an expression that resolves to the key. If the property has no value, the portlet does not sort the tabs.

Sort Order (order)
Indicates the order in which to sort the tabs. Specify an expression that resolves to the direction. If the property has no value, the portlet sorts the tabs in ascending order.

Include Items (includeItems)
Indicates whether to include items as tabs or to only include folders. Specify one of the following:

- `true` – In addition to folders, also include items as tabs.
- `false` – Include only folders as tabs.

Simple Property Editors Component

Portlet Title	Simple Property Editors
Portlet Name	wm_simplepropertyeditors
Portlet File Name	wm_simplepropertyeditors.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.simplepropertyeditors
Default Instances of the portlet	None

Property Editors. Page developers use this component to deploy a collection of legacy property editors. Page developers can use these editors when developing portlet views. Each editor provides a specialized user interface for editing a portlet preference on the portlet Properties page.

Legacy Property Editors that are Deployed with the Component

Property Editor	Description
boolean	Renders a check box whose value is true when the check box is selected; otherwise the value is false.
button	Renders a button.
checkbox	Renders a check box.
checkbox-multiple	Renders a set of check boxes.
file	Renders a hidden input field.
imagebutton	Renders an image button.
java.lang.Object	Renders a text input field. This is the same as the text property editor.
label	Renders the value of the property as a simple label.
label_with_fallback	Renders the property as a simple label if the property has a value; otherwise the label to render is retrieved from a resource bundle.
list	Renders a multi-select list box.
localizableLabel	Renders a label that is retrieved from a resource bundle.
map	Renders a panel that an end user can use to enter name/value pairs.

Property Editor	Description
orderedmap	Renders a panel that an end user can use to enter name/value pairs. The end user can also order the name/value pairs.
password	Renders a password input field.
password-confirm	Renders a password input field with validation to ensure the passwords match.
radio	Renders a radio button.
radio-multiple	Renders a set of radio buttons.
select	Renders a list box from which a user can select one entry; that is, a select-one list box.
select-categorized	Renders a select-one list box whose entries are categorized.
select-multiple	Renders a list box from which a user can select multiple entries; that is, a select-many list box.
submit	Renders a submit button.
text	Renders a text input field.
textarea	Renders a multi-line text input field.
textarea-popup	Renders a multi-line text input field where a user edits the property value via a popup window.

Spotlight Tab Styles Portlet

Portlet Title	Spotlight Tab Styles
Portlet Name	wm_tabstyles_spotlight
Portlet File Name	wm_tabstyles_spotlight.pdp
Top-level Folder	default
JSR168 Portlet?	No
Alias	portlet.tabstyles.spotlight
Default Instances of the portlet	None

Page Components. Page developers indirectly use this portlet when using the Dynamic Tabs Portlet (wm_tabs) to display tabs as links with the select link displayed in the skin's "spotlight" style. For example, in some skins the "spotlight" style for a link might be outlined with an oval background.

Properties that are not Displayed in the User Interface

layout
Indicates whether to display the tab links horizontally or vertically. Specify one of the following:

- `horizontal` – Default. Display a horizontal list of links.
- `vertical` – Display a vertical list of links.

selectAll
Indicates whether to display all of the links as though all of the links were selected. Specify one of the following:

- `true` – Display all of the links as though all of the links were selected.
- `false` – Default. Display only the selected link.

Properties Configured by the Tabs Portlet

The Tabs (wm_tabs) portlet automatically configures the following properties to provide the raw list of tabs to display in a popup style.

Titles (titles)
Defines the list of titles to use for the tabs. If the property has no value, no tabs are displayed when the user clicks on or hovers over the target text.

Tooltips (tooltips)
Defines the list of tool tips for the tabs. If the property has no value, no tool tips are displayed.

Links (links)
Defines the list of links to use for the tabs. If the property has no value, the tabs will not have links.

Next (next)
Identifies the "Next" link used to display the next page of tabs. If the property has no value, a "Next" link is not displayed.

Previous (prev)
Identifies the "Previous" link used to display the previous page of tabs. If the property has no value, a "Previous" link is not displayed.

Selected (selected)
Identifies the index of the tab that is to be selected when rendering the tabs. When **Selected** is `-1`, no tab is selected. The default is `-1`.

Tab Styles Portlet

Portlet Title	Tab Styles
Portlet Name	wm_tabstyles

Portlet File Name	wm_tabstyles.pdp
Top-level Folder	template
JSR168 Portlet?	No
Alias	portlet.tabstyles
Default Instances of the portlet	None

Template. Page designers use this portlet to dynamically create tabs that use a standard style.

Properties

Titles (titles)
Defines the titles of the tabs. Specify a comma-separated list of titles. If the property has no value, no tabs will be displayed.

Tooltips (tooltips)
Defines the tooltips for the tabs. Specify a comma-separated list of tooltips. If the property has no value, the tabs will not have tooltips.

Links (links)
Defines the links to use for the tabs. Specify a comma-separated list of links. If the property has no value, the tabs will not be linked.

Next (next)
Defines the link to use for the **Next** button. Specify the URL of the page to display when the end user clicks the **Next** button. If the property has no value, the **Next** button is not displayed.

Previous (previous)
Defines the link to use for the **Previous** button. Specify the URL of the page to display when the end user clicks the **Previous** button. If the property has no value, the **Previous** button is not displayed.

Selected (selected)
Defines the first selected tab. Specify the index of the selected tab. If the property has no value, or is not a valid index, no tab is selected.

Toggle Opened/Closed Portlet

Portlet Title	Toggle Opened/Closed
Portlet Name	wm_toggle
Portlet File Name	wm_toggle.pdp
Top-level Folder	extras
JSR168 Portlet?	No

Alias	/portlet/wm_toggle
Default Instances of the portlet	None

Page components. End users use this portlet to toggle the visibility of an element in another portlet. The portlet is intended for use only with legacy portlets for which toggling capabilities are not available. It it not needed for newer portlets because the webMethods CAF toolkit provides several methods for toggling whether controls are visible or hidden.

A page developer adds the Toggle Opened/Closed portlet to a page and configures it to identify the element to toggle. The portlet displays a toggle icon that the end user uses to toggle the display of the element.

Properties

Target IDRef (targetId)
Identifies the element for which to toggle the display. Specify the ID of an element in another portlet. If this property has no value, the Toggle Opened/Closed portlet does not display the toggle icon.

Direction Opened (directionOpened)
Indicates the direction that the toggle icon, an open arrow, should point. Specify one of the following:

- `r` – Default. Display the open arrow pointing to the right.
- `l` – Display the open arrow pointing to the left.
- `u` – Display the open arrow pointing up.
- `d` – Display the open arrow pointing down.

Initially Opened (initiallyOpened)
Indicates whether to initially make the toggled element visible or hidden. Specify one of the following:

- `true` – Default. Initially open the toggled element so that it is visible.
- `false` – Initially close the toggled element so that it is hidden.

Save In Cookie (saveInCookie)
Indicates whether to save the current toggle state in a cookie sot that the next time the page is displayed, the saved toggle state is used. Specify one of the following:

- `true` – Save the toggle state in a cookie. If the saved state is hidden, the next time the page is displayed the element is hidden; if the saved state is visible, the next time the page is displayed the element is visible.
- `false` – Default. Do not save the toggle state.

Cache Age (cacheAge)

Indicates how long, in minutes, to keep the content of the portlet in cache. All instances of the portlet share the value of this property. Specify one of the following:

- `0` – Default. Do not cache the content.
- *n* – Cache the content for the specified number of minutes (*n*).
- `-1` – Cache the content indefinitely.

Universal Picker Portlet

Portlet Title	Universal Picker
Portlet Name	wm_universalpicker
Portlet File Name	wm_universalpicker.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.universalpicker
Default Instances of the portlet	None

System. Portlet developers use this portlet to display a picker list of My webMethods Server items from which an end user can select one or more of the items.

Typically, this portlet is used only as a property editor for the properties of another legacy portlet. For new webMethods CAF portlets, the Portal Resource Picker Dialog control provides this same ability.

Properties

Multi-Select (multiple)
Indicates whether the picker allows an end user to select multiple entries. Specify one of the following:

- `true` – Default. End users can select multiple entries.
- `false` – End users can select only a single entry.

Type (type)
Identifies types of resources that an end user can select with the picker. Specify one or more of the following:

- `all` (default)
- `portlet`
- `folder`
- `link`

- `user`
- `group`
- `role`
- any portlet name
- any DBO type name

Use a comma-separated list to specify more than one type.

Default Roots (defaultRoot)
Identifies the default set of roots (container IDs that appear as root folders) in the My webMethods Server taxonomy to display. Specify a comma-separated list of thingIDs or aliases to identify the roots. The default value is `folder.root,folder.public,user.current.home`, which means that the default set of roots under which the user can browse for items to select is the My webMethods Server root folder (named Folders), the public folder (named Public Folders), and the current user's home folder (named My Folders).

Roots (roots)
Identifies the set of roots (container IDs that appear as root folders) in the My webMethods Server taxonomy to display. Specify a comma-separated list of thingIDs or aliases to identify the roots. If the property has no value, the portlet uses the value of the Default Roots (defaultRoot) property.

Default Selected (selected)
Identifies the entries that should be initially selected. Specify a comma-separated list of thingIDs or aliases to identify the set of selected entries. If the property has no value, no entries will be initially selected.

Selectable Script (selectable)
Defines how to determine whether an entry is selectable. Specify a JavaScript expression that the portlet evaluates to determine whether to allow the end user to select an entry. By default, end users can select all entries.

Unselectable Script (unselectable)
Defines how to determine whether an entry can be deselected. Specify a JavaScript expression that the portlet evaluates to determine whether to allow the end user to deselect an entry. By default, end users can deselect all entries.

8 My webMethods Server Components Used Internally

This chapter describes the portlets and components that My webMethods Server uses internally.

Summary of Portlets and Components

The following table lists the portlets and components that My webMethods Server uses internally.

Name	Description
Designer Shell	Tools. This shell is the default shell for the Preview Server, which is a miniature instance of My webMethods Server hosted inside Software AG Designer that enables developers to preview portlets as they build them.
Email Deliverer Portlet	System. The system uses this portlet to deliver My webMethods Server and task e-mail notifications using the SMTP protocol.
Pseudo Portlet Portlet	System. Administrators use this portlet to display the portlet view for any My webMethods Server thing that does not have any other portlet view configured for it. The Pseudo Portlet portlet displays a view that includes the icon for the My webMethods Server thing, its name and description, and a link to the thing.
Shortcut View Portlet	System. End users implicitly use this portlet when accessing a shortcut, which is a reference to an object (e.g., folder, page, portlet) in My webMethods Server. This portlet is part of the core system components. Users should not attempt to publish or use a standalone Shortcut View portlet.
Thumbnail Agent Portlet	Thumbnailer. The system uses this portlet to create thumbnail images when end users select to display a thumbnail view of a My webMethods Server folder. This is a system portlet that cannot be published by an administrator, page developer, nor end user.
wm_cafshared Web Application	System. The system uses this web application to serve some secondary resources (for example, JavaScripts, cascading style sheets, and images) that are needed CAF controls to function.

Designer Shell

Component Name	wm_shell_designer
Component File Name	wm_shell_designer.cdp
Top-level Folder	development

Tools. This shell is the default shell for the Preview Server, which is a miniature instance of My webMethods Server hosted inside Software AG Designer that enables developers to preview portlets as they build them.

This wm_shell_designer component is based on the Blank shell, but includes the following two additions.

- A link to open the SOAP Monitor Portlet
- The Locale Switcher Portlet

Email Deliverer Portlet

Portlet Title	Email Deliverer
Portlet Name	wm_emaildeliverer
Portlet File Name	wm_emaildeliverer.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

System. The system uses this portlet to deliver My webMethods Server and task e-mail notifications using the SMTP protocol. The Email Deliverer portlet expects that the mail session is available via JNDI for your task application.

Typically you should not need to change the default values of this portlet's properties, which define the number of SMTP threads to allocate for e-mail delivery and the number of recipients per e-mail. However, if the defaults do not meet the needs of your system, an administrator can modify the properties. To display the properties:

1. Log into My webMethods Server using the sysadmin user.
2. Navigate to **Folders > System > Extended Types > portlets > System**.
3. In the list, locate and open the wm_emaildeliverer object.

Properties

Number of SMTP Threads (smtpWorkerPoolSize)
Defines the number of SMTP threads to allocate for delivering e-mail messages. Specify a positive whole number. The default is `6`.

Even if you only define only `1` SMTP thread, the portlet will eventually deliver all e-mail notifications although it might take significant time. You might want to define more than the default `6` threads if delivery of e-mail notifications is a priority and if the system is heavily used for My webMethods Server and task e-mail notifications.

Max number of recipients per e-mail (maxRecipientsPerEmail)
Defines the maximum number of recipients to receive a single e-mail message. The portlet can deliver an e-mail notification to more than one e-mail address, for example, when multiple users subscribe to a notification. Specify a positive whole number. The default is `100`.

Note:
This property defines the maximum number of SMTP addresses to be delivered; it is *not* the number of addresses on the e-mail TO line.

Queue Name (queueName)
Defines the name of the JMS queue that the portlet uses to request e-mail delivery. This property is not displayed in the portlet's user interface. It is set to `com.webmethods.portal.notification.email.msg.queue.messages`. Do not modify the property.

Pseudo Portlet Portlet

Portlet Title	Pseudo Portlet
Portlet Name	wm_psuedoportlet
Portlet File Name	wm_psuedoportlet.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	portlet.pseudoportlet
Default Instances of the portlet	None

System. Administrators use this portlet to display the portlet view for any My webMethods Server thing that does not have any other portlet view configured for it. The Pseudo Portlet portlet displays a view that includes the icon for the My webMethods Server thing, its name and description, and a link to the thing.

Note:
Administrators can configure My webMethods Server so that it uses an alternative portlet instead of the Pseudo Portlet portlet. To do, the administrator changes the portlet.pseudoportlet alias to point to a different portlet.

Properties

Item (item)
Identifies the thing to display. Specify the My webMethods Server ID of the thing. If the property has no value, the portlet displays the current page.

Cache Content (cacheAge)
Defines how long to cache the portlet's contents. Specify one of the following:

- `-1` – Default. Cache the contents of the portlet indefinitely.
- `0` – Do not cache the contents of the portlet.
- *number* – Cache the contents of the portlet for the specified number of minutes.

Shortcut View Portlet

Portlet Title	Shortcut View
Portlet Name	wm_shortcutview
Portlet File Name	wm_shortcutview.pdp
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

System. End users implicitly use this portlet when accessing a shortcut, which is a reference to an object (e.g., folder, page, portlet) in My webMethods Server. This portlet is part of the core system components. Users should not attempt to publish or use a standalone Shortcut View portlet.

Note:
Users can create shortcuts when editing a page or using the Publish Portlet. When configuring a shortcut, the user specifies the object to which the shortcut points.

Thumbnail Agent Portlet

Portlet Title	Thumbnail Agent
Portlet Name	wm_thumbnailagent
Portlet File Name	wm_thumbnailagent.pdp
Top-level Folder	default
JSR168 Portlet?	No

Alias	/portlet/wm_thumbnailagent
Default Instances of the portlet	None

Thumbnailer. The system uses this portlet to create thumbnail images when end users select to display a thumbnail view of a My webMethods Server folder. This is a system portlet that cannot be published by an administrator, page developer, nor end user.

An administrator can deploy the Thumbnail Agent portlet to any My webMethods Server folder. After it is deployed, when using the Folder View portlet, an end user can request the thumbnail view. Currently, this portlet only creates thumbnail images for JPEG, GIF, and BPM image files. Creating thumbnails images for HTML files, My webMethods Server pages, and portlets is deprecated because it requires the use of an external web service.

Thumbnail Generation Params Properties

Thing to thumbnail (generateThumbnail_url)
Identifies the file for which to create a thumbnail image. Specify the alias or URI of a published My webMethods Server JPEG, GIF, or BPM file. If the property has no value, the portlet does nothing.

Thumbnail Width (generateThumbnail_width)
Defines the width of the thumbnail image to create. Specify the number of pixels for the width. The default value is `128`.

Thumbnail Height (generateThumbnail_height)
Defines the height of the thumbnail image to create. Specify the number of pixels for the height. The default value is `128`.

wm_cafshared Web Application

Portlet Title	wm_cafshared
Portlet Name	wm_cafshared
Portlet File Name	wm_cafshared.war
Top-level Folder	system
JSR168 Portlet?	No
Alias	None
Default Instances of the portlet	None

System. The system uses this web application to serve some secondary resources (for example, JavaScripts, cascading style sheets, and images) that are needed CAF controls to function.

Note:

When deploying CAF web applications and portlets to a non-My webMethods Server container, be sure to deploy the wm_cafshared web application to the container.

Init Parameters

VERSION_KEY
Identifies the system property that holds the wm_cafshared web application version string. Specify the name of the system property. The default is `wm_cafshared.version`.

The wm_cafshared web application calculates the version string from the timestamp in its manifest file. When rendering links to resources that are served by the wm_cafshared web application, the webMethods CAF uses the version string to ensure that the client browsers retrieve new versions of the resources when the version string is updated.

If this parameter has no value, webMethods CAF does not include the version string on links to wm_cafshared resources, and as a result, a client browser might use a older, cached version of a resource.

DEBUG
Indicates whether the wm_cafshared web application should serve the "debug" version of JavaScript resources. The debug version includes comments and white space from the JavaScript files. Specify one of the following:

- `true` – Serve the "debug" version, leaving comments and white space from the JavaScript files.
- `false` – Default. Do *not* serve the "debug" version, and as a result, strip comments and white space from the JavaScript files.