OZONE Widget Framework

Administrator's Guide

September 28, 2012

Publication/Revision History

Release	Date
Revised – OWF 6	September 28, 2012
Revised – OWF 5	June 29, 2012
Initial Document – OWF 4.0	March 30, 2012

Contents

Tables

Figures

1 Introduction

1.1 Objectives

This guide covers topics relevant to administering the OZONE Widget Framework (OWF).

1.2 Document Scope

This guide is intended for administrators of an OWF instance. An administrator can, for the purpose of this guide, be defined as someone who maintains OWF, as well as handles user and widget settings.

1.3 OWF Purpose

OWF is a set of tools, generally delivered in the **OWF-bundle-6-GA.zip**. When deployed, OWF is used for organizing and displaying Web applications (Widgets) in a single browser window.

1.4 Supported Browsers

OWF supports Internet Explorer 7 and higher and Firefox 3.6 and higher. OWF is tested against the following browsers:

Table 1: Tested Browsers

Browsers	Versions
Internet Explorer	7 & 9
Firefox	3.6 & 13

1.5 Related Documents

Table 2: Related Documents

Document	Purpose
User's Guide	Understanding the **OWF user interface**; adding, deleting, modifying **widgets** and using widget **intents**; accessing and using **Marketplace**; creating, deleting, adding, switching, modifying **dashboards**, defining accessibility features such as **high-contrast themes and keyboard navigation**
Administrator's Guide	Understanding **administrative tools**: adding, deleting, and editing widgets, users, groups, and group dashboards; **creating default content** for users, groups and group dashboards
Developer's Guide	**Creating Widget applications and integrating existing applications into OWF**; widget upgrade instructions; walkthroughs for creating widgets; adding the following components to widgets: intents, descriptor URLs, preference API; logging and launching API
Configuration Guide	Overview of **basic architecture and security**; OWF **installation** instructions; defining and instructions for modifying default settings; **database** set up and logging guidance; **framework and theme** customization instructions; OWF **upgrade instructions**, directions for adding and deleting **help** content
Quick Start Guide	Walkthrough of **basic OWF functions** such as using widgets and dashboards; instructions for **setting up a local instance of OWF**, unpacking the OWF bundle and installing **security certificates**; Truststore/Keystore changes

2 Administrator Tools

Administration tools, located by clicking the administration button on the toolbar, allow an administrator to manage group dashboards, widgets, groups, users, and widget approvals from Marketplace.
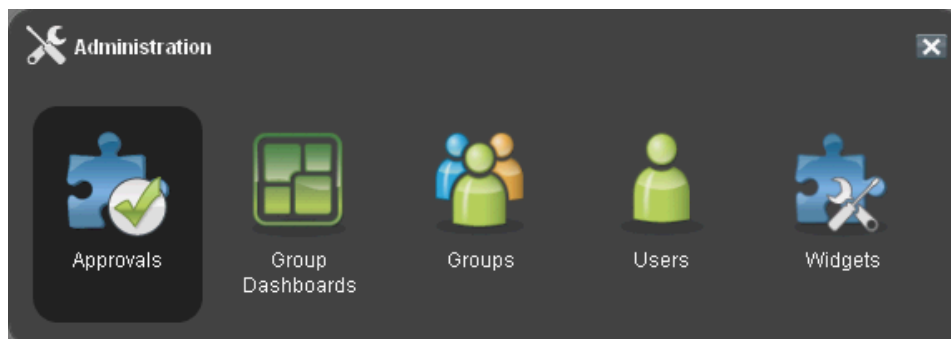
Figure 1: Administrator Tools

2.1 Administrative Managers

The administration managers are used to create, edit, and delete users, groups, widgets, and group dashboards as well as approve listings imported from Marketplaces. While each manager has specific fields that relate to the manager's specific purpose, some of the functions operate identically in each manager. For example, the search feature in the user manager functions exactly like the search feature in the widget manager. Accordingly, search is explained only once in this section.

Also, this document no longer contains definitions regarding basic information that general users should understand. If a topic can be easily defined by Google, it has been removed from this guide. The following section offers a general overview of the administrative managers and their use.

The manager information is broken into sub-sections:

- Panel
- Management buttons
- Search bar and pagination toolbar

2.1.1 Manager Panel
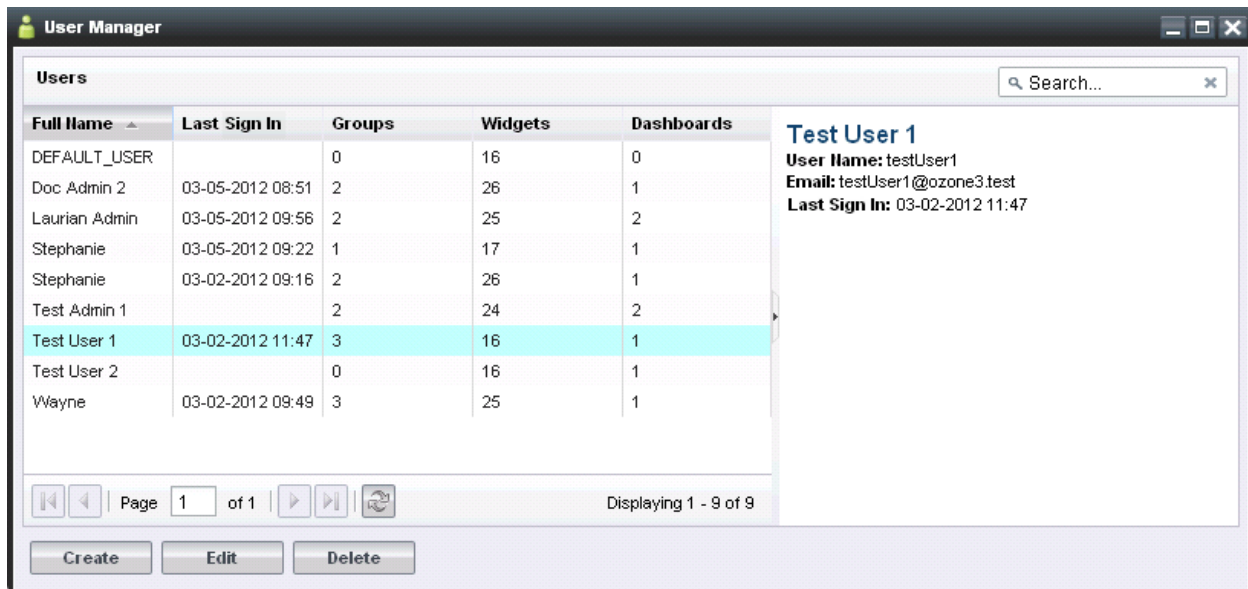
Figure 2: User Manager Panel

The dashboard, widget, group, and user managers open to similar panel views. The panel view described in this section applies to all four managers.

The Panel:

- Allows the user to create, edit, delete, or view an entry.

- Displays the number of users/groups/widgets/dashboards associated with the specific entry.

  Note: When viewing the widget count, only the widgets that a user requests or receives from an administrator appear in the overall count. Widgets associated through groups will NOT appear in the widget count.

- Offers a view of the first fifty results in alphabetical order. Additional results can be viewed using the pagination as described in section 2.1.3. To reduce the number of displayed results: Use the search bar, described in section 2.1.2.

From the panel, an administrator can:

- **Sort -** Most of the columns in the panel can be sorted in ascending or descending order by clicking on the triangle to the right of the column header and selecting a sorting option.

- **Hide/Show columns -** Columns can be hidden or shown by hovering over a column header, clicking the triangle that will appear, hovering over the columns menu option, and un-checking the columns to be hidden.

- **Reorder columns -** Columns can be reordered by clicking (and holding) a column header down and then dragging it to the desired position.

- **Multiple selection** - Entries can be selected for bulk operations by holding down the CTRL button while clicking multiple entries. The delete, edit, activate, and deactivate buttons will perform bulk operations on all selected entries.

- **View the information panel -** To display more information about the entry, single-click the row to open the information panel on the right.

2.1.1.1 Management Buttons: Create/Delete

Administrators use the manager widgets to create and delete users, widgets, groups, and group dashboards. Differences between the four managers are referenced in sub-bullets.

Create - Launches the editor widget. From the editor, an administrator can create a new group/user/group dashboard/widget (depending on which editor the administrator launches) and assign related data to the new entry.

- When creating a new entry, only the properties tab will be active until the administrator saves the group/user/group dashboard/widget via the apply button.

Delete - Deletes selected entries. Some group/user/group dashboard/widget rules apply:

- Deleting a **group** does not delete the users or widgets assigned to the group. It only deletes the *pairing* of users with widgets in the group.

- Deleting a **widget** removes it from a user's launch menu and the groups to which it was assigned.

- Deleting a **group dashboard** removes it from the group. Individual users can continue to use their copy of the dashboard. However, the user's individual copy will no longer have the option to restore to the group dashboard as described in section 3.2.

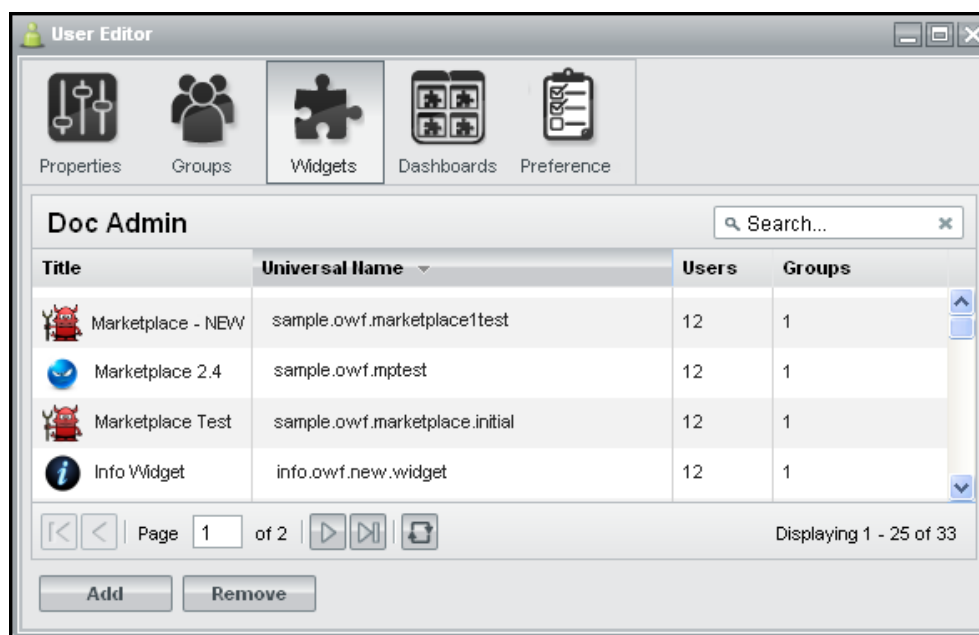**Edit** - When a group/user/group dashboard/widget is selected, clicking edit launches the editor widget, which allows an administrator to edit the entry, as described in the next section.

2.1.1.2 Management Buttons: Edit

Clicking edit on any administrative manager will launch the respective group/user/group dashboard/widget editor.
Note: If an administrator launches the editor widgets in a Fit Dashboard, the editor

widgets will "float" on top of the dashboard. Additional dashboard layout information is found in the User's Guide.

Figure 3: Widgets Tab on the User Editor

From the Editor Widgets, administrators can create, edit and delete data assigned to group/user/group dashboard/widget. The following table specifies widget properties. The last column of the table indicates where the field will be encountered.

Table 3: Fields

Field	Purpose	Location
Activate/Deactivate	Users in active groups have full access to their group-assigned widgets. Users in a deactivated group will not have access to any of the widgets which are assigned to them via the deactivated group. When a group becomes deactivated it will appear gray. *Note: If a user is in Group A and Group B and each group has Widget 1 assigned to it, the user will still have access to Widget 1 if Group A is deactivated and Group B is activated. Additionally, if the user has widget access outside of a group's distribution, the user will not lose access to it, even if they lose group access.*	Group manager (under the split edit button); "Active" checkbox in the Group Editor, Properties Tab
Background	Some widgets do not have user interfaces. These widgets are often used to cache or log data. If a widget is set to run in the	Widget Editor, Properties Tab

Field	Purpose	Location
	background, it will not appear in the foreground of the OWF dashboard. However, it will appear in the Widget Switcher (Alt + Shift + Q). Also, it will appear on the user's launch menu if the "visible" menu flag (described in this table) is turned on.	
Container Icon URL	Defines the location of the icon which appears in widget chrome at 24x24 pixels.	Widget Editor, Properties Tab
Copy To Group	Allows an administrator to copy dashboards to selected groups via the add groups window. Once a dashboard is added to a group, every member of that group will receive their own copy of that dashboard.	User Editor, Dashboards Tab Edit split-button
Default Tags	Specifies the default tags (comma separated, if needed) that facilitate widget categorization. Default tags cannot be deleted by the user; however, users can add additional tags to widgets in their instance of OWF. Default Tags are superseded by grouped widgets, which are explained in section 3.6.	Widget Editor, Properties Tab
Definition	A required field in a dashboard that contains the JSON configuration of a dashboard.	Dashboard Editor, Properties Tab
Display Name	The group name which will appear in grids and tables throughout administrator views.	Group Editor, Properties Tab
GUID	A unique 32-character alpha-numeric code for a particular named Widget. If "Widget A" is launched 5 times, all five widgets will share the same `widgetGuid` property.	All Editor Widgets, Properties Tab
Height	Defines the launch height of the widget in pixels. Up and down arrows to the right of the field can be used to modify the overall height.	Widget Editor, Properties Tab
Intents	Widget intents build on OWF's publish/subscribe functionality by allowing users to choose the widget that will use its data. Intents explain the intention for the widget. This binding capability enables two widgets to enhance each other's functionality. *Note: Only developers can modify intents via the*	Widget Editor, Intents Tab

Field	Purpose	Location
	widget's descriptor file; instructions are available in the OWF Developer's Guide.	
Launch Menu Icon URL	Defines the location of the icon which appears in the launch menu (at 128 x 128 pixels), provided the "visible" menu flag (mentioned below) is checked.	Widget Editor, Properties Tab
Remove	Separates the selected user/group/widget/dashboard from the selected entry. This does not delete the user/group/widget/dashboard from the system. It only removes the assignment to the selected entry.	All Editors
Singleton	Designates whether a widget can only have one instance launched per dashboard.	Widget Editor, Properties Tab
Universal Name	A global identifier which is a permanent element of the widget across all instances of OWF. This differs from a widget GUID which is unique to a specific installation. *Note: This field will only be visible in the Widget Editor if a developer has specified a Universal Name in an imported Widget Descriptor URL.*	Widget Editor, Properties Tab
URL	Defines the location of the web application to which the widget icon will link. This is a required field.	Widget Editor, Properties Tab
User Management	Defines whether or not the group is an automatic group, being populated and maintained by external sources. This value cannot be modified once the group has been created.	Group Editor, Properties Tab
Version	Displays the version number of the listing. This is completely user-driven and is for informational purposes.	Widget Editor, Properties Tab
Visible	Dictates whether a listing will show in a user's launch menu. This cannot be overridden by the user.	Widget Editor, Properties Tab
Widget Type	A drop-down menu for selecting the widget type which determines where the widget will be located. Starting with OWF 4.0, only	Widget Editor, Properties Tab

Field	Purpose	Location
	standard widgets appear in the launch menu. Administration widgets will appear under the administration button on the toolbar. Widgets set to type Marketplace will appear under the Marketplace button on the toolbar. Metric widgets appear under the Metric button on the toolbar. *Note: Instructions explaining how to associate OWF with a Metrics Service are found in the OWF Configuration Guide.*	
Width	Defines the launch width of the widget in pixels. Up and down arrows to the right of the field can be used to modify the overall width.	Widget Editor, Properties Tab

2.1.2 Manager Widgets—Search

 - Reduces the entries displayed in the panel to entries containing the specified word or characters entered in the search bar. Clicking the X button will clear the filter and display all entries in the panel. Clicking the search magnifying glass button will apply the search and display the filtered results in the panel.

Note: This is a full-text search and it is NOT case-sensitive.

2.1.3 Manager Widgets—Pagination

Figure 4: Management Widget Pagination Toolbar

- **-** Navigates between pages of results displayed in the search results panel.

- **-** Refreshes the results in the search results panel, maintaining the current filtering and sorting options.

- **-** Displays the number of results being shown against the overall total in the system.
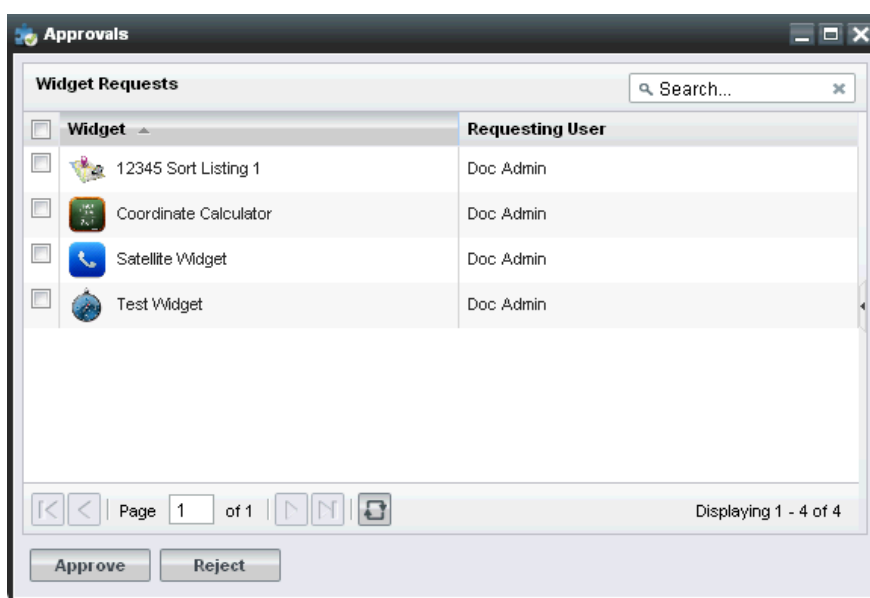
2.2   Widget Approval

When a user adds a widget from Marketplace, the widget is stored in a pending state until approved by an administrator. To approve a widget, an administrator must navigate to the OWF approval window by clicking the administration button on the toolbar and then choose the approvals button.
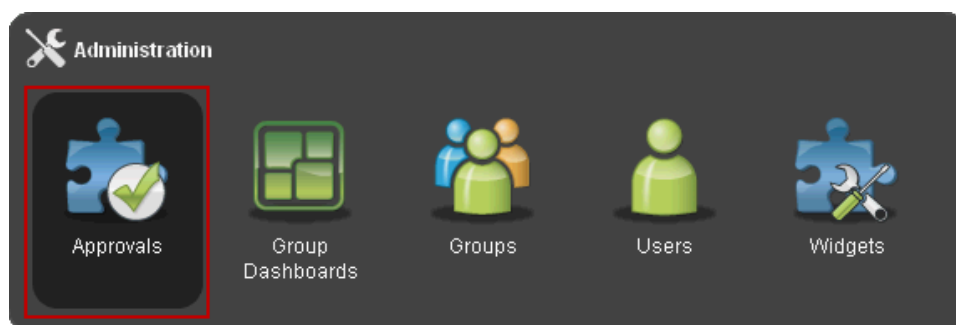
Figure 5: Marketplace Approval

The approvals widget will open, which lists all the widgets pending approval. The list can be sorted by widget or requesting user. A widget will not be available to a user until an administrator approves it and the user refreshes their browser.

Figure 6: Approvals Widget

To approve or reject widgets:

1) Check the checkbox to the left of a pending widget or widgets.
   Note: Widgets must be approved or rejected for each user. Approving a widget for one user will not approve it for another.

2) Click approve or reject at the bottom of the window.

3) After approving the widget, it will appear in the requesting user's launch menu. If a widget is rejected, it will be removed from the widget approval dialog.

2.3 Approving Required Widgets

A Marketplace listing can require other Marketplace listings. For example, if a user requests Widget A and it requires Widget B, the user automatically requests widgets A and B. This relationship is further explained in the OWF User's Guide.

In the Approvals Widget, an administrator has to ways to identify that a Marketplace listing requires other Marketplace listings. When a listing is selected:

- The details section of the listing will display: **Requires Widgets: true**

- Its requirements will appear below the listing details

Both identifiers are highlighted in the following example:
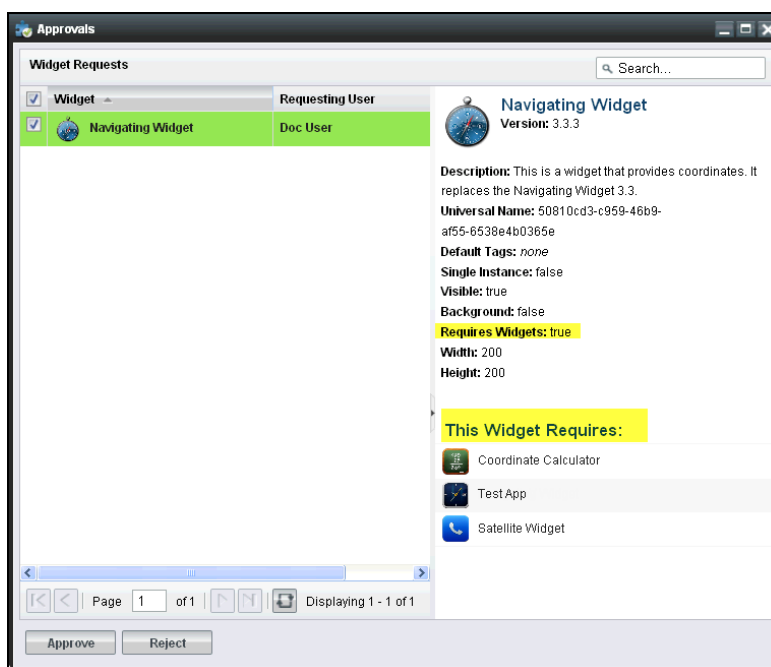
Figure 7: Required Widget Identifiers

If an administrator approves a widget that requires other widgets, the required widgets will be automatically approved.

3   Adding and Organizing Content

Adding and organizing users' widgets, dashboards and groups will be explained in this section. OWF ships with a default user and default group. Administrators can use the default user profile and default group to add widgets and dashboards to a user or several users' instance of OWF.

DEFAULT_USER – A default user profile that ships with OWF. The DEFAULT_USER data will automatically be assigned to every new user of a particular OWF installation.

When a new user enters OWF for the first time, the DEFAULT_USER data will be applied and copied to that user's profile. After the initial login, any changes that the user makes will only impact their data from that point on. The DEFAULT_USER data remains unchanged and will continue to be applied to all new users.

Note: If an administrator makes changes to the DEFAULT_USER data set, it will only impact the users who log in for the first time, following the change. Any users who received the data prior to the change will not be affected.

OWF Users – A default user group that ships with OWF. Every new user is automatically assigned to it.

When a new user enters OWF for the first time, the OWF Users group data will be applied and copied to that user's profile. After the initial login, any changes that the user makes will only impact their data from that point on. However, if an administrator changes the OWF Users Group, the change will be applied to all users who have access to the group.

3.1   Adding DEFAULT_USER Content

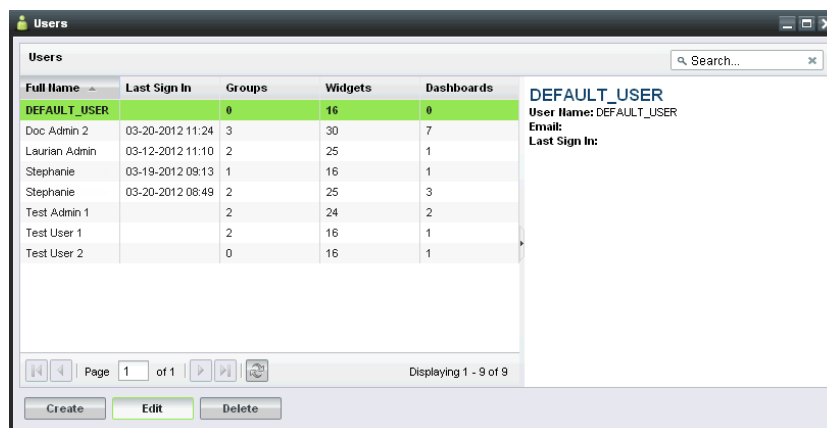The DEFAULT_USER is available in the User Manager shown below:

Figure 8: Add User Dialog/DEFAULT_USER

For the purposes of administration, the DEFAULT_ USER should be treated as a regular user and administrator's can add, delete or edit the DEFAULT_USER content in the same way that they modify any other user's data as explained in the following section.

3.1.1 Adding Widgets to a User's Account

The following serves as a walkthrough for documentation purposes only. To add widgets to a user's account:

1) Click the button on the toolbar.

2) Click the users button to open the user manager.

3) From the manager, select a user. Then, click edit to open the user editor.

4) Click the widget tab at the top of the editor. The widgets that are already associated with the user will display in the window.

5) To add widgets, click the add button. A modal window will display all widgets that are available to that user. Select widgets, then, click the OK button. The widget will be automatically added to the list of widgets on the user's widget tab.

Use this basic formula to add groups and dashboards via the appropriate tabs in the user editor widget. When assignments are complete, all of the data which has been applied to the DEFAULT_USER will be instantly available for all new users of OWF.

3.2 Editing Existing User and Widget Content

To edit existing user and widget content:

1) Click administration on the toolbar, select widgets or users to open the respective manager.

2) From the manager, select the widget or user account that needs to be updated and click edit.

3) Update data on the properties tab and click apply. Update each field, then click apply. For definitions of less common fields, see section Table 3: Fields.

   Note: Administrators cannot change property fields or intents for widgets that use Descriptor URLs. Changes to these fields must be made in the descriptor files, see the OWF Developers Guide for more information.

3.3 Creating Widgets

Starting in OWF 4.0, administrators can use the widget type function to separate widgets on a user's toolbar. Only standard widgets will appear in the launch menu.

Administration widgets will appear under the administration button on the toolbar. Widgets set to type Marketplace will appear under the Marketplace button on the toolbar. Widgets set to type Metric will appear under the Metric button on the toolbar.

To create a widget:

1) Click the ![icon] button on the toolbar.

2) Click the widgets button to open the widget manager.

3) From the manager, click the create button to open the widget editor.

4) From here there are two ways to create widgets:

    a. **Use a Descriptor URL**—Enter a Descriptor URL and click Load. For more information about Descriptor URLs see section 3.3.1: Widget Descriptor URL.

    b. **Manually enter data**—If a Descriptor URL is not available, click the "Don't have a descriptor URL?" link and complete the fields on the properties tab. Remember that the widget type will dictate the location of the widget. For more information about specific entry fields, see Table 3: Fields.

5) Click apply. This will unlock the users and groups tabs on the widget editor. Select each tab and add users and groups to the widget.
   Note: The intents tab lists relationships between widgets. These widgets are established by a developer. This tab is for informational purposes and not editable via the OWF interface.

6) Refresh OWF. The new widget should appear under the respective toolbar button. For more details about connecting to Marketplace(s), see section 3.3.2 Connecting to Marketplace.

3.3.1   Widget Descriptor URL

Descriptor URLs allow users to create widgets without entering the widget's information manually. The user simply enters a URL and the widget's information is automatically retrieved from a descriptor file that a developer maintains. Administrators **cannot** change properties for widgets that use Descriptor URLs.

Descriptor URLs offer several benefits. They reduce the risk of typing errors when entering widget data. They allow for several installations of OWF to easily share widget information via the descriptor file. In addition, Descriptor URLs provide each widget with a universal name, a unique identifier which is a permanent element of the widget across all instances of OWF.

Also, descriptor URLs provide a mechanism for developers to add widget intents. Widget intents are the instructions for carrying out a widget's intentions. Intents build on OWF's publish/subscribe functionality by allowing users to choose the widget that

will use its data. Intents explain the intention for the widget. This binding capability enables two widgets to enhance each other's functionality.

Note: Find instructions about using intents in the OWF User's Guide and instructions about creating intents in the OWF Developer's Guide.

3.3.2 Connecting to Marketplace

Marketplace, similar to a commercial application store, operates as a thin-client registry of applications and services. Marketplace allows administrators to connect to multiple Marketplaces by creating widgets that are stored under the Marketplace button on the toolbar.

To enable the Marketplace button on the toolbar, an administrator has to create a Marketplace widget. To do this, follow these instructions:

1. Sign in to OWF as an administrator.

2. Click the ⚒ button on the toolbar and select widgets.

3. Click create and complete all the fields, **change the widget type to Marketplace**.

4. Click apply.

5. When you refresh Marketplace, the Marketplace button will appear on the toolbar. Repeat the process for each different Marketplace connection.

3.4 Creating Groups and Adding Users/Widgets

Groups allow administrators to easily add or remove widgets and dashboards to and from multiple users' instances of OWF. For example, by putting 10 employees into a group, the administrator can send 1 dashboard or 1 widget to the group, rather than having to individually add the widget to 10 employees.

The following serves as a walkthrough for documentation purposes only. To create groups in OWF and to add users to that group:

1) Click the ⚒ button on the toolbar.

2) Click the groups button to open the groups manager.

3) From the manager, click the create button. This will launch the group editor.

4) Fill in the appropriate data on the properties tab. Once the apply button is clicked, the other tabs on the group editor will become active. Note the following:

    a. If the display name field is populated, that is the name which will appear in all administrator panels. The only way for the primary name field to appear is to leave the display name field blank.

    b. If an administrator checks the automatic box in the user management section, the following will happen:

        i. The name of the group will not be editable.

        ii. The users tab will remain grayed out as users can only be added via external sources, such as a security plugin.

5) Add users to the group by selecting the users tab, clicking the add button, and selecting users from the grid.

6) Widgets and Dashboard tabs behave the same way and can be used to populate groups accordingly.

Note: OWF ships with two automatic groups: OWF Administrators and OWF Users. New members will be added to them in accordance with their respective role. These groups cannot be deleted or renamed. In the event that either group is single-selected, the delete button will be grayed out. If either (or both) groups are selected along with other manual groups, the delete button will be active. However, upon clicking delete, only the manual groups will be removed from the system.

3.5 Creating Group Dashboards

Group dashboards allow members of a group to have identical copies of a dashboard. Similar to the OWF Users Group, administrators can use group dashboards to give different groups of users the same starting screen or access to widgets. Each user can customize their instance of the group dashboard. Those changes will ONLY affect that user's instance of the dashboard. Users can return their dashboard to the *current state* of the group dashboard by clicking the restore button (found by clicking the settings button on the Toolbar, then choosing the dashboard button, and choosing restore from the triangle to the right of the edit button). If an administrator changed the group dashboard after it was added to a user's instance of OWF the *current state* of the dashboard may be different than the one that originally appeared on the user's navigation bar.

Note: If a group dashboard is deleted by an administrator, the users' copies of that dashboard will remain available to that user. However, the restore feature will no longer be available for the dashboard.

To create a group dashboard:

1) Click the ⚒ button on the toolbar.

2) Open the group dashboards manager.

    a. Select an existing dashboard and click edit.

    b. Click create to make a new dashboard.

3) Customize the fields on the properties tab, then click apply.

4) Select the groups tab, click add to select groups

5) Close the window. Each user in the selected group will see the dashboard in the dashboard switcher on their toolbar.

3.6   Grouping Widget

Administrators can use grouped widgets, group dashboards and default tags to categorize and separate widgets on users' launch menus. Administrators add widgets to a dashboard and then give group(s) access to that dashboard to group widgets via the group dashboard feature explained in section 3.3. Administrators also use groups and default tags to group widgets.

3.6.1   Adding Widgets to a Group

When a user is part of a group, for example Group A, the user's launch menu will display the widgets assigned to Group A in the main section of the launch menu. A list of the user's group displays in the left panel. By checking the box to the left of the group name, the launch menu and favorites pane will only display widgets that are associated with the group, as shown below. Only an administrator can make changes to widgets that are part of a group. Users cannot edit or delete the group or the widgets in it.
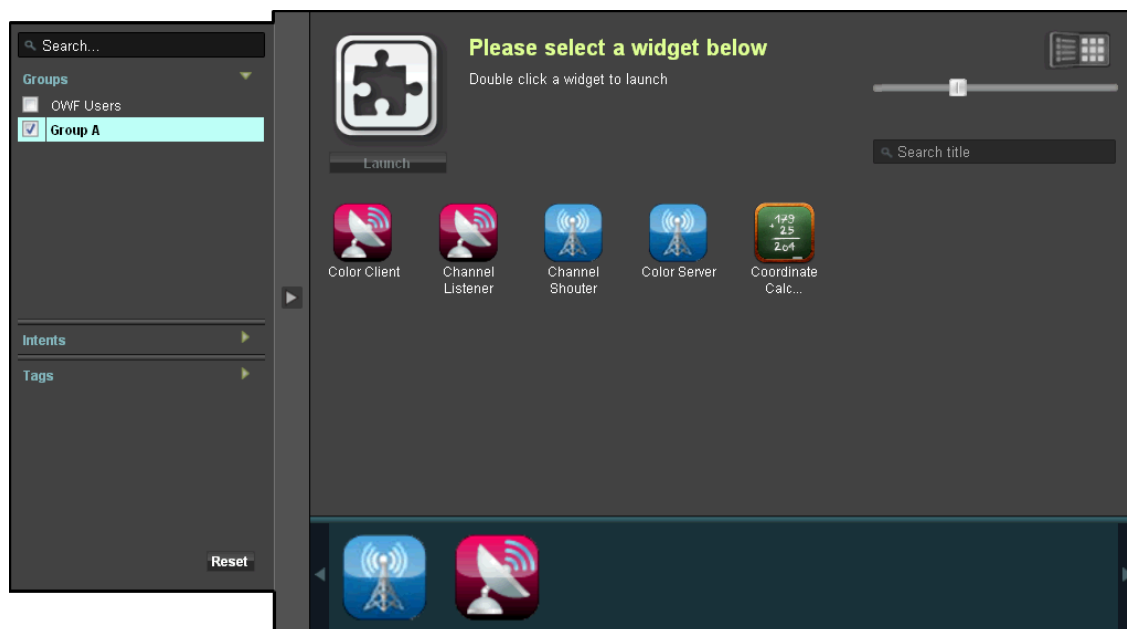
Figure 9: Grouped Widgets

To add widgets to a group:

1) Click the ⚒ button on the toolbar.

2) Click the groups button.

    a. Select an existing group and click edit.

3) The group editor will launch.

    a. Select the widgets tab.

    b. If it's a new group, populate the properties information (field descriptions found in Table 3), and click apply. Then, click the widgets tab.

4) Click add to select widgets that will be assigned to the group, then click OK.

5) Close the group editor.

3.6.2 Adding Default Tags

The previous sections explained how to add widgets and users to a group. Administrators can add default tags that will categorize widgets on all users' launch menus. To add a default tag:

1) Click the button on the toolbar.

2) Click the widget button to launch the widget manager. Then, select a widget.

3) Click edit to open the widget editor.

4) Add a default tag, then click apply.

5) Close the widget edit window. In every user's launch menu, the widget will appear in the newly created default tag category.

In addition to default tags, users can add their own tags to widgets. The user-applied tags will only appear in the user's instance of OWF.

Note: Default tags are not applied to administration, Marketplace, or metrics widgets. Also, if a default tag is added to a widget after the widget is assigned to a user (visible in their launch menu), the default tag will not apply unless the widget is deleted from the user's instance and reassigned to the user.

3.7 Preferences
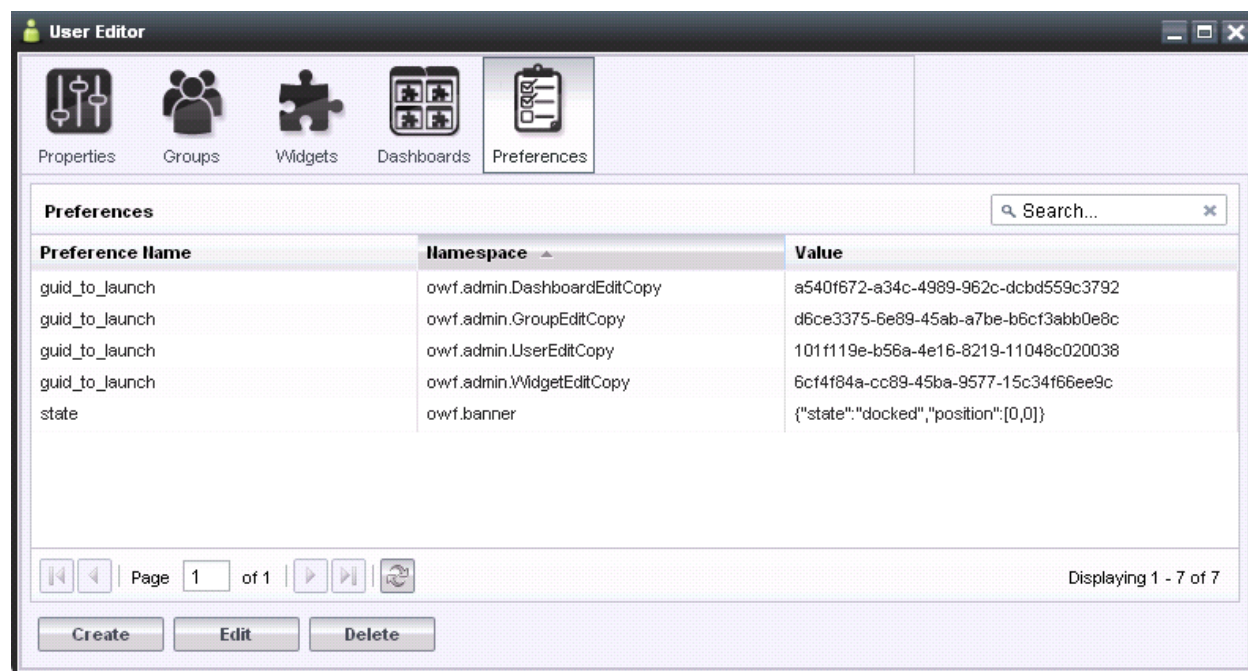
Figure 10: Preferences Tab

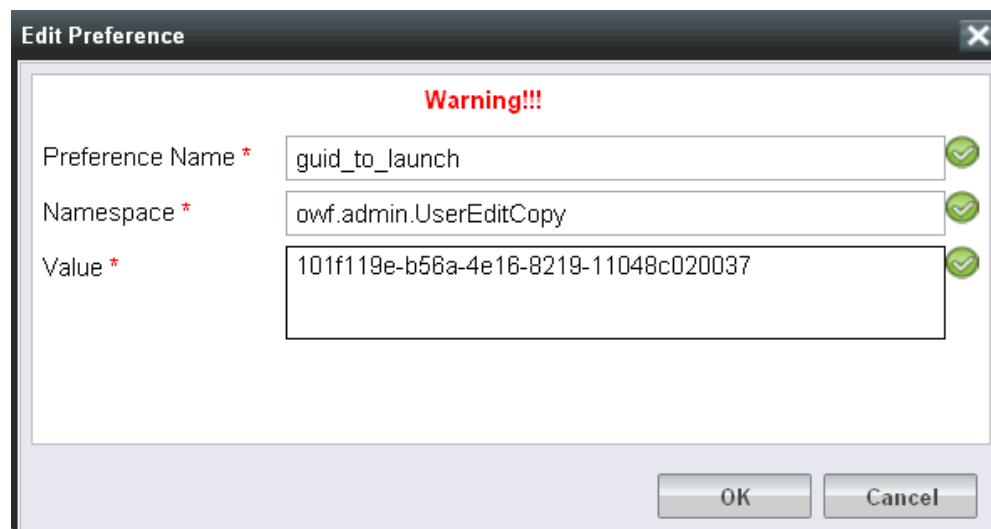Widgets use preferences to store data. Preferences include widget location on the screen, launching instructions, etc. A preference value can be created and saved any time a widget performs an action. Once that preference is saved to the database, it will appear on the preferences tab in the user editor. The preference tab serves as a table of known preferences. From the tab, administrators can view, create, edit, and delete preferences. However, from the preference tab, the administrator cannot configure a widget to use a preference. For a widget to respond to preferences, the widget must be configured via the preference API.

If a widget is configured to use preferences, they can be used to define screen location, widget interaction, etc. For example, the `guid_to_launch` preference is a useful eventing tool. Administrators can use it to make a tracking widget launch a map widget. In the Figure 11 example, the `guid_to_launch` preference causes the User Manager to launch a copy of the User Editor.

The preferences tab includes the following fields:

- **Preference Name -** The preference name is referred to as the "key" for the preference item. It lists the name of the preference as dictated by the widget or OWF. If the widget uses preferences, OWF will add the preference name to the table on the preference tab whenever the action that is associated with the preference is performed.

- **Namespace -** The namespace is the identifier for the widget or system category. Generally these identifiers will describe general functionality for a widget or set of widgets.

- **Value -** Stored inside of the preference, values house the data that the preference uses. An example could be the actual widget GUID value that the preference will use to launch a widget. This is a string value but developers can use **JSON** or REST URIs as the preference value.

Figure 11: Preferences Dialog

Appendix A   Object Reference

A.1   Dashboard Object

Table 4: Dashboard Object

Property	Sample Value	Description
guid	1d789781-e4e5-5acb-f9c5-eae28adf7699 (A unique 32-character alpha-numeric code)	A unique identifier for the dashboard. *Note: The guid is only required to be unique within the dashboard table.*
columnCount	0	Deprecated
isdefault	True/False	Legacy code. This is no longer applicable.
layoutConfig	'height', 'cls', 'items'	Holds the various panes on a dashboard and the widgets inside of the panes, including the widget states.
layout	N/A	Deprecated
EDashboardLayoutList	N/A	Deprecated
name	Customizable	This is a User-driven field. Even if a Dashboard loads a default name, the User can modify it at will.
state	N/A	Deprecated
groups	Customizable	When a group is associated with the dashboard, it becomes the value for the "Groups" attribute.
isGroupDashboard	True/False	Identifies if the dashboard is associated with a group.
description	Customizable	A description of the dashboard.
defaultSettings	N/A	Deprecated
createdDate	02/09/2012 11:48	Date when the dashboard was created.

Property	Sample Value	Description
	AM EST	
editedDate	02/09/2012 11:48 AM EST	Date when the dashboard was last edited.
createdBy	userId: "" userRealName: ""	The user name and id of the dashboard creator.
user	Customizable	When a name is used to validate a security protocol, it becomes the value for the "User" attribute.
alteredByAdmin	True/False	If a User's Dashboard has been modified via the Administration UI, this value will appear as true.

A.2   Widget Object

Table 5: Widget Object

Property	Sample Value	Description	Dashboard Layout Type: Accordion, Tabbed, Desktop, Portal, Fit
uniqueId	1d789781-b4a5-512b-ea59-eae28adf7699	A unique 32-character alpha-numeric code. representing a Widget's state as it exists on a Dashboard instance	All Dashboards which contain widgets *Note: This value is not associated with a widget's state within a Dashboard until the Widget is launched – Then the uniqueId is created.*
widgetGuid	1a718777-b4a5-512b-ea59-eae28adf7699 (A unique per widget 32-character alpha-numeric code)	A unique 32-character alpha-numeric code for a particular named Widget. If "Widget A" is launched 5 times, all five widgets will share the same widgetGuid.	All Widgets
universalName	Ozone.OWF.Cha	A user-generated,	All Widgets

Property	Sample Value	Description	Dashboard Layout Type: Accordion, Tabbed, Desktop, Portal, Fit
	nnelShouter	custom identifier which is a permanent element of the widget across multiple instances. This differs from a widgetGuid which is unique to a specific installation.	
widgetVersion	1.0	A string which indicates the software version of the Widget.	All Widgets
name	"Widget B"	Up to 50 alpha numeric characters.	All Widgets
active	True/False	Dictates which particular Widget on a Dashboard is active upon launch.	Tabbed Desktop
width	225	Widget width in pixels.	All Widgets
height	400	Widget height in pixels.	All Widgets
minimized	True/False	Dictates whether a Widget is at its minimized size.	Tabbed Desktop Floating Widgets
maximized	True/False	Dictates whether a Widget is at its maximized size.	Tabbed Desktop Floating Widgets
pinned	True/False	Denotes whether a Widget is pinned in place or not.	Accordion
collapsed	True/False	Denotes whether a Widget is collapsed to its chrome. Dashboard types (on right) are what differentiate this attribute from the minimized definition.	Portal Accordion

Property	Sample Value	Description	Dashboard Layout Type: Accordion, Tabbed, Desktop, Portal, Fit
floatingWidget	True/False	Denotes whether a Widget is floating over an entire dashboard and not constrained to a single layout type. Example: Widgets launched from a Widget in a Fit pane open as floating widgets.	Fit
Column	N/A	Deprecated	N/A
buttonID	N/A	Deprecated	N/A
buttonOpened	N/A	Deprecated	N/A
region	N/A	Deprecated	N/A

The following three values are stored in the database for stateful purposes but are not part of the widget definition or widget creation process. They are used to determine widget placement on a dashboard.

Table 6: Widget Object Placement

Property	Sample Value	Description	Dashboard Layout Type :Accordion, Tabbed, Desktop, Portal, Fit
X	200	The 'X' value represents the physical location (from the upper left-hand corner of OWF) in which the Widget will spawn.	Tabbed Desktop Floating Widgets
y	500	The 'Y' value represents the physical location (from the upper left corner of OWF) in which the Widget will spawn.	Tabbed Desktop Floating Widgets
zIndex	Variable number	A relative number representing which Widget is "on top" of a stack of Widgets – thus making it the visible Widget.	Tabbed Desktop Floating Widgets

A.3 User Preference Object

Table 7: Preference Object

Property	Sample Value	Purpose
Preference Name	ClockWidgetSettings	User driven – defines the overall preference description. This is the effective key of the preference object that identifies the specific preference.
namespace	TimeType	Used to uniquely identify a Widget preference. The OWF team recommends using namespaces similar to java packages, i.e. com.mycompany.widgetname.
value	24hrMilitary	This value is used to store individual preferences that will be recalled by Widget developers. It may consist of any value from a string to a JSON object or REST URI.
userId	testUser1	The unique identifier of the user to whom this preference belongs.

Appendix B   Known Issues

B.1   Browser Issues

Launching any of the Editors (as well as many JavaScript-heavy widgets) in certain versions of Internet Explorer consumes system memory that won't be flushed or released until Internet Explorer is exited and restarted. Again, this is currently only an Internet Explorer issue.

B.2   User Interface Issues

Changes in screen resolution may render widgets unviewable.

The positioning of the Widgets is absolute. This means that when changing from a larger monitor to a smaller monitor, or when changing from a higher screen resolution to a lower screen resolution, some floating windows may be either partially or fully off the viewable region of the screen. Currently there is no remedy for this issue.

Internet Explorer users may experience degraded performance.

The latest release of Sencha's ExtJS 4 JavaScript framework is known to have performance issues in Internet Explorer browsers. The Sencha team is actively working to release a patch to address these issues. Once the patch is released, the OWF team will integrate it into the codebase.

Appendix C   Contact Information

C.1   Discussion Group

The OZONE Developers Discussion Group is hosted through Google Groups at http://groups.google.com/group/ozone-developers. This forum is for the distribution of release announcements, Q&A related to OWF and for additional inquiries about widgets and features being developed across the user base. Access to the group can be gained by requesting an invitation at http://groups.google.com/group/ozone-developers or by contacting the Community Support Team at goss-support@owfgoss.org.

C.2   Additional POCs

For information about the OZONE Widget Framework or access to its resources, please email goss-support@owfgoss.org. Additional resources can be found at http://owfgoss.org.