

**Oracle® WebCenter Sites**

User's Guide for the Community Application

11g Release 1 (11.1.1)

February 2012

Oracle® WebCenter Sites User's Guide for the Community Application, 11g Release 1 (11.1.1)

Copyright © 2012 Oracle and/or its affiliates. All rights reserved.

Primary Author: Melinda Rubenau

Contributing Author: Tatiana Kolubayev

Contributor: Julia Pererva, Gennady Svetlichiny, Alex Vushkan, Aleksey Zvolinsky

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle America, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle and Java are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information on content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services.

## Table of Contents

<b>About This Guide</b> .....	<b>9</b>
Audience .....	9
Related Documents .....	10
Conventions Used in This Guide .....	10
Quick Steps .....	10
Terms and Definitions .....	10
Third-Party Libraries .....	10
<b>1 Welcome to the Oracle WebCenter Sites: Community Application</b> .....	<b>11</b>
Overview .....	12
Types of Community Widget Tags .....	15
Deployment Options .....	16
Community Interface .....	17
Community Application Roles and Permissions .....	18
<b>2 Getting Started</b> .....	<b>25</b>
Logging In .....	26
How Do I Deploy Community Widget Tags?.....	28
Next Steps .....	33
<b>3 Working with the Comments Widgets</b> .....	<b>35</b>
Overview .....	36
Moderating Comments .....	36
Reconfiguring the Comments Widget's Default Settings .....	47
Deploying the Comments Widget Tags .....	52
Deploying a Default Comments Widget Tag .....	53
Deploying Custom Comments Widget Tags .....	56
Custom Widget Tag Deployment Screens .....	58

<b>4 Working with the Reviews Widgets</b> .....	<b>67</b>
Overview .....	68
Moderating Reviews .....	68
Reconfiguring the Reviews Widget's Default Settings .....	79
Deploying the Reviews Widget Tags .....	84
Deploying a Default Reviews Widget Tag .....	85
Deploying Custom Reviews Widget Tags .....	89
Custom Widget Tag Deployment Screens .....	92
<b>5 Working with Ratings Widgets</b> .....	<b>103</b>
Overview .....	104
Moderating Ratings .....	104
Reconfiguring the Ratings Widget's Default Settings .....	113
Deploying the Ratings Widgets .....	116
Deploying a Default Ratings Widget Tag .....	117
Deploying Custom Ratings Widget Tags .....	120
Custom Ratings Widget Tag Deployment Screens .....	123
<b>6 Working with the Login Bar Widget</b> .....	<b>137</b>
Overview .....	138
Reconfiguring the Login Bar Widget's Default Settings .....	138
Deploying a Login Bar Widget Tag .....	140
<b>7 Working with Polls</b> .....	<b>143</b>
Overview .....	144
Working with the 'All Polls' Screen .....	144
Creating a New Poll .....	146
<b>8 Reconfiguring General Community Widget Settings</b> .....	<b>151</b>
Overview .....	152
Working with the 'Settings' Menu .....	152
<b>9 Working with the Cache Tool</b> .....	<b>161</b>
Overview .....	162
Viewing Cache Information .....	162
Viewing Information in the Common Cache .....	165
Viewing Information in the Dependencies Cache .....	167
Viewing Information in the Shared Cache .....	169
<b>10 Testing Deployed Community Widget Tags</b> .....	<b>171</b>
Sample Site with Community Widgets Deployed .....	172
Working with Comments and Reviews Widgets as an Authenticated Visitor .....	173
Logging in to the Site as an Authenticated Visitor .....	173

Posting Comments as an Authenticated Visitor . . . . .	174
Posting Reviews as an Authenticated Visitor . . . . .	176
Working with Comments as an Authenticated Visitor . . . . .	177
Modifying Comments on the Website . . . . .	177
Deleting Comments from the Website . . . . .	178
Working with Reviews as an Authenticated Visitor . . . . .	179
Modifying Reviews on the Website . . . . .	179
Deleting Reviews from the Website . . . . .	180
Working with Comments and Reviews Widgets as an Anonymous Visitor . . . . .	181
Posting Comments as an Anonymous Visitor . . . . .	181
Posting Reviews as an Anonymous Visitor . . . . .	182
Sorting Comments and Reviews . . . . .	183
Flagging Inappropriate Posts . . . . .	184
Replying to Posted Comments . . . . .	185
Viewing Review Details . . . . .	186
Ranking Visitor Comments and Reviews . . . . .	187
Sharing Comments and Reviews with Facebook and Twitter . . . . .	187
Creating and Sharing a Permalink . . . . .	188
Working with Ratings Widgets . . . . .	189
Working with Poll Widgets . . . . .	190
<b>A. Analyzing Community Widget Tags . . . . .</b>	<b>193</b>
Comments Widget Tags . . . . .	194
Comments Widget Tag . . . . .	194
Comments Summary Widget Tag . . . . .	197
Links to Topics Widget Tag . . . . .	200
Recently Commented Widget Tag . . . . .	203
Most Commented Widget Tag . . . . .	206
Reviews Widget Tags . . . . .	209
Reviews Widget Tag . . . . .	209
Reviews Summary Widget Tag . . . . .	212
Links to Topics Widget Tag . . . . .	215
Reviews Average Rating Widget Tag . . . . .	218
Top Ranked Reviews Widget Tag . . . . .	222
Recently Reviewed Widget Tag . . . . .	225
Most Reviewed Widget Tag . . . . .	228
Ratings Widget Tags . . . . .	231
Stars Ratings Widget Tag . . . . .	231
Thumbs Up/Down Ratings Widget Tag . . . . .	234
Like It Ratings Widget Tag . . . . .	237
Recommend Ratings Widget Tag . . . . .	240
Ratings Average Rating Widget Tag . . . . .	243
Recently Rated Widget Tag . . . . .	247
Most Rated Widget Tag . . . . .	250

---

Login Bar Widget Tag .....	253
Poll Widget Tags .....	256
Main Poll Widget Tag.....	256
Poll Results Widget Tag.....	259
<b>B. Enabling SEO Support for Community Widgets .....</b>	<b>263</b>
Overview .....	264
Enabling SEO Support .....	265
<b>C. Deploying the CSS Tag .....</b>	<b>267</b>
Overview .....	268
Deploying the CSS Tag.....	268

## List of Tables

Table 1: Community Widget Tags .....	15
Table 2: Community Application Roles and Permissions .....	20
Table 3: Visitor Permissions to Community Widgets on the Website .....	22
Table 4: Moderating All Comments .....	37
Table 5: Moderating Comments by Topic .....	40
Table 6: Moderating Comments by User .....	44
Table 7: Reconfiguring the Comments Widget's Appearance Settings .....	48
Table 8: Reconfiguring Commenting Permissions for Site Visitors .....	51
Table 9: Deploying the Comments Widget Tags .....	52
Table 10: Customizing and Deploying the Comments Widget Tag .....	59
Table 11: Customizing and Deploying the Comments Summary Widget Tag .....	61
Table 12: Customizing and Deploying the Recently Commented Widget Tag .....	62
Table 13: Customizing and Deploying the Most Commented Widget Tag .....	64
Table 14: Moderating All Reviews .....	69
Table 15: Moderating Reviews by Topic .....	72
Table 16: Moderating Reviews by User .....	76
Table 17: Reconfiguring the Reviews Widget's Appearance Settings .....	80
Table 18: Reconfiguring Reviewing Permissions for Site Visitors .....	83
Table 19: Deploying the Reviews Widget Tags .....	84
Table 20: Customizing and Deploying the Reviews Widget Tag .....	93
Table 21: Customizing and Deploying the Reviews Summary Widget Tag .....	95
Table 22: Customizing and Deploying the Average Rating Widget Tag .....	96
Table 23: Customizing and Deploying the Top Ranked Reviews Widget Tag .....	97
Table 24: Customizing and Deploying the Recently Reviewed Widget Tag .....	99
Table 25: Customizing and Deploying the Most Reviewed Widget Tag .....	101
Table 26: Moderating All Ratings .....	105
Table 27: Moderating Ratings by Topic .....	108
Table 28: Moderating Ratings by User .....	110
Table 29: Reconfiguring the Ratings Widget's Appearance Settings .....	113
Table 30: Reconfiguring Rating Permissions for Site Visitors .....	115

---

Table 31: Deploying the Ratings Widget Tags .....	116
Table 32: Customizing and Deploying the Stars Ratings Widget Tag .....	124
Table 33: Customizing and Deploying the Thumbs Up/Down Ratings Widget Tag ..	126
Table 34: Customizing and Deploying the Like It Ratings Widget Tag .....	128
Table 35: Customizing and Deploying the Recommend Ratings Widget Tag .....	130
Table 36: Customizing and Deploying the Average Rating Widget Tag .....	131
Table 37: Customizing and Deploying the Recently Rated Widget Tag .....	132
Table 38: Customizing and Deploying the Most Rated Widget Tag .....	134
Table 39: Reconfiguring the Login Bar .....	139
Table 40: Working with the “All Polls” screen .....	144
Table 41: Creating a Poll .....	146
Table 42: Authorizing a Domain .....	153
Table 43: Configuring Restricted Words .....	154
Table 44: Configuring the Visitor Blacklist .....	156
Table 45: Configuring the Visitor Whitelist .....	157
Table 46: Configuring Moderation Settings for Widgets .....	158
Table 47: Overview of Columns in the Summary Screen .....	164
Table 48: Overview of Columns in the Common Cache Screen .....	165
Table 49: Overview of Columns in the Dependencies Cache screen .....	167
Table 50: Overview of Columns in the Shared Cache screen .....	169

---

## About This Guide

This guide describes Oracle WebCenter Sites: Community, a social computing application designed to gather visitors' comments, reviews, and ratings on website content. The Community application also enables its users to create and manage polls, which can be used to survey site visitors about desired topics. This guide begins with an overview of the Community application and its users. It continues to the process of configuring and deploying widgets on web pages, and describes methods for moderating site visitors' comments, reviews, and ratings, as well as managing poll results.

Applications discussed in this guide are former FatWire products. Naming conventions are the following:

- *Oracle WebCenter Sites* is the current name of the application previously known as *FatWire Content Server*. In this guide, *Oracle WebCenter Sites* is also called *WebCenter Sites*.
- *Oracle WebCenter Sites: Community* is the current name of the application previously known as *FatWire Community Server*. In this guide, *Oracle WebCenter Sites: Community* is also called *Community*.
- *Oracle WebCenter Sites: Community Blogs* is the current name of the application previously known as the *FatWire Community Server Blog Module*. In this guide, *Oracle WebCenter Sites: Community Blogs* is also called *Community Blogs*.

The Community application integrates with Oracle WebCenter Sites according to specifications in the *Oracle WebCenter Sites 11g Release 1 (11.1.1.x) Certification Matrix*. For additional information, see the release notes for the Community application. Check the WebCenter Sites documentation site regularly for updates to the *Certification Matrix* and release notes.

## Audience

This guide is for Community application administrators, moderators, and designers. Users are assumed to have a comprehensive knowledge of their company's business needs and a basic understanding of their role in site development and management processes. All Community application users should have a clear understanding of the content on their company's websites in order to effectively moderate visitor feedback.

Administrators and designers should also have experience with the WebCenter Sites Admin interface, and must be familiar with modifying template assets and publishing those templates to the website.

## Related Documents

For more information, see the following documents:

- *Oracle WebCenter Sites Administrator's, Developer's, and User's Guide for Community Blogs*
- *Oracle WebCenter Sites Administrator's Guide*

## Conventions Used in This Guide

The following text conventions are used in this guide:

- **Boldface** type indicates graphical user interface elements that you select.
- *Italic* type indicates book titles, emphasis, or variables for which you supply particular values.
- `Monospace` type indicates file names, URLs, sample code, or text that appears on the screen.
- **Monospace bold** type indicates a command.

## Quick Steps

Many instructions in this guide are written as “quick steps” to provide readers with a quick reference for accessing various Community interface screens. For example, the instruction for accessing the Comments widget tag’s “Deployment” screen reads as follows:

Select **Comments > Deploy > Comments**

The step above means:

Select the **Comments** option, select **Deploy**, and then select **Comments**.

When features and associated operations require explanation, the steps are written in detail.

## Terms and Definitions

This guide uses terms that are specific to the Community application such as *Community Widget Tag*. These terms are explained in the “Glossary” section at the end of this guide.

## Third-Party Libraries

Oracle WebCenter Sites and its applications include third-party libraries. For additional information, see *Oracle WebCenter Sites 11gR1: Third-Party Licenses*.

## Chapter 1

# Welcome to the Oracle WebCenter Sites: Community Application

This chapter introduces the Oracle WebCenter Sites: Community application and the community widget tags that you can deploy on your websites to collect visitor feedback.

This chapter contains the following sections:

- [Overview](#)
- [Types of Community Widget Tags](#)
- [Deployment Options](#)
- [Community Interface](#)
- [Community Application Roles and Permissions](#)

## Overview

- *You have published new content to one of your company's websites and you wish to collect feedback from site visitors.*
- *You and your staff want to know which site content is most popular among visitors. Using community widgets, visitors are able to post comments, rate content, and write reviews, and you are able to better understand your visitors' preferences.*
- *Your website needs a moderator and a variety of options for managing visitors' comments, reviews, and ratings. With the Community application you can allow your site visitors to post comments, reviews, and ratings automatically to the website, or you can monitor visitors' input before displaying it on the website.*
- *You want to conduct visitor surveys on your website. The Community application enables you to create customized polls which can be used to gather feedback from your visitors to better understand their preferences on a given topic.*
- *You want to add corporate blogs to your website. If you install the Community Blogs module on a content management site, it provides the site with default components that enable you to add blog functionality to the website.*

Community is a social computing application that runs on the Web Experience Management (WEM) Framework. When the Community application is enabled on a content management (CM) site, it provides configurable community widget tags that can be deployed on any web pages to display comment, review, rating, poll, and login bar widgets to visitors. Site visitors can form communities by utilizing these widgets to lead discussions, express opinions, participate in polls, and share experiences about specific topics on the sites. Administrators and moderators of the Community application can collect input from such visitors in order to better understand visitors' preferences and adjust site content to their expectations.

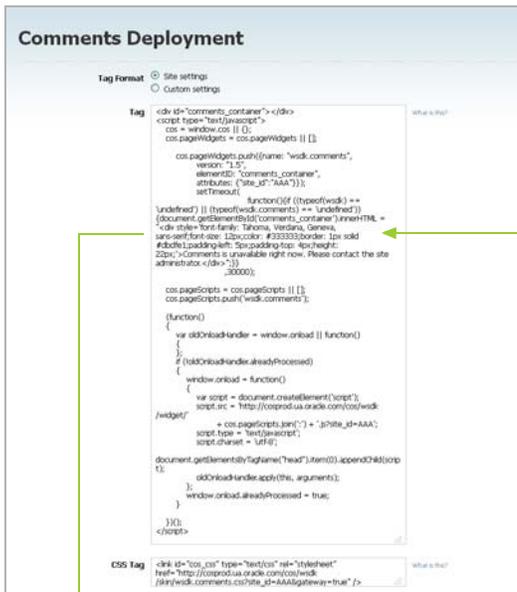
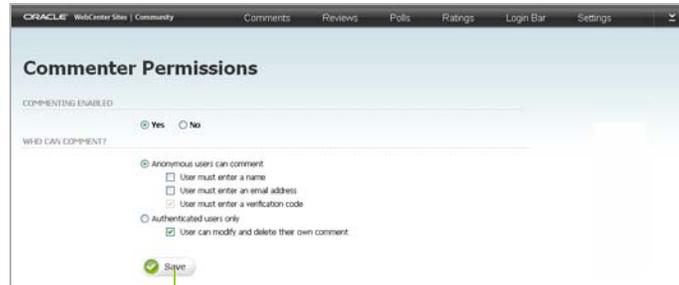
All comments, reviews, and ratings posted to the website are displayed in the Community interface where they can be manually moderated and/or automatically moderated by the auto-moderation filters ("Restricted Words," "Visitor Blacklist," and/or "Visitor Whitelist"). Poll instances and their associated results are also displayed in the Community interface.

[Figure 1, on page 13](#) illustrates how a default widget tag's settings are reconfigured in the Community interface and deployed on a web page. Our example uses the Comments widget tag (for a complete list of the widget tags the Community application supports, see [Table 1, on page 15](#)). [Figure 2, on page 16](#) presents a simplified diagram of a Community application installation, illustrating the events that occur when you configure and deploy community widget tags.

The Community application also supports blog functionality. Integrating the Oracle WebCenter Sites: Community Blogs module with the Community application enables your content providers to publish corporate blogs that display the Community application's comment fields and links. For more information about the Community Blogs module, see the *Oracle WebCenter Sites Administrator's, Developer's, and User's Guide for Community Blogs*.

**Figure 1:** Overview of reconfiguring and deploying default community widget tag settings

1. Default community widget tag settings are reconfigured by using the desired widget's "Permissions" and "Appearance" configuration screens. Once saved, the settings are also saved to the widget's "Tag" field in its deployment screen.

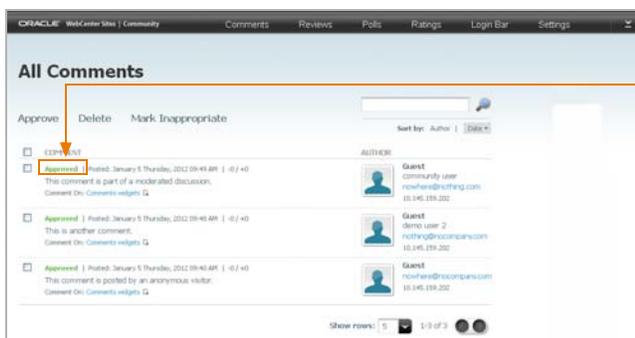


2. The widget defined in the "Tag" field is displayed on web pages once the tag is deployed on those pages.



**Comments widget displayed on a web page.**

3. Comments posted to the website are also displayed in the Community interface's comments moderation screens (**Comments > Moderate**). This example shows the comments displayed in the "All Comments" screen (**Comments > Moderate > All Comments**).



If moderation is enabled, then comments require approval in the Community interface. If moderation is disabled, then comments are automatically approved and displayed on the website.

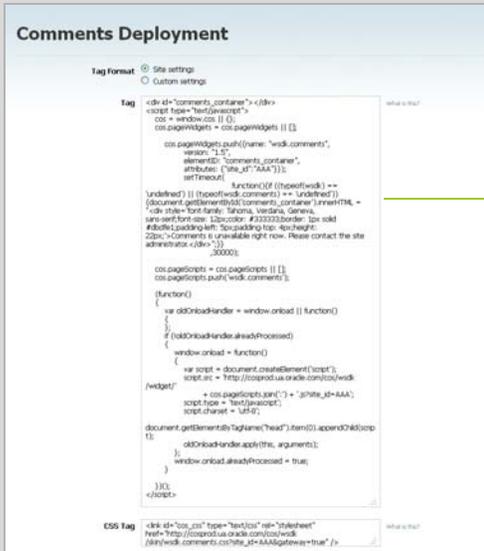
### A Closer Look at Widget Tags

1. Each type of widget is defined by its own tag and each tag is associated with its own deployment screen in the Community interface (on the site on which the Community application is enabled).

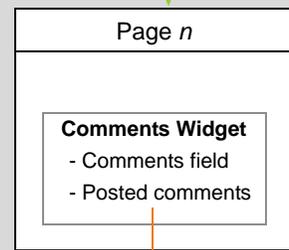
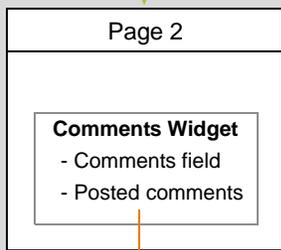
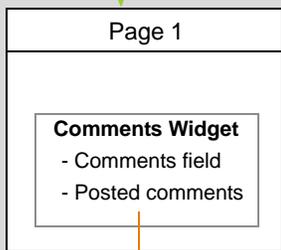
For example, the Comments widget tag is located in the “Comments Deployment” screen and defines:

- The look and feel of the widget.
- The production (delivery) system, which serves the widget.
- The CM site (hosted by the production and management systems) where the Community application is enabled, which is the site where comments will be collected (see Figure 2).

Each widget tag defines similar properties. The types of widgets you can work with in the Community interface are listed in Table 1, on page 15.



2. The widget tag can be deployed on any web pages by administrators or designers.



3. All visitors' comments are collected on the production CM site (defined in the widget tag) and displayed in the Community interface, where they can be moderated (if moderation is enabled).



## Types of Community Widget Tags

The following table lists the community widget tags administrators and designers can deploy on any web pages:

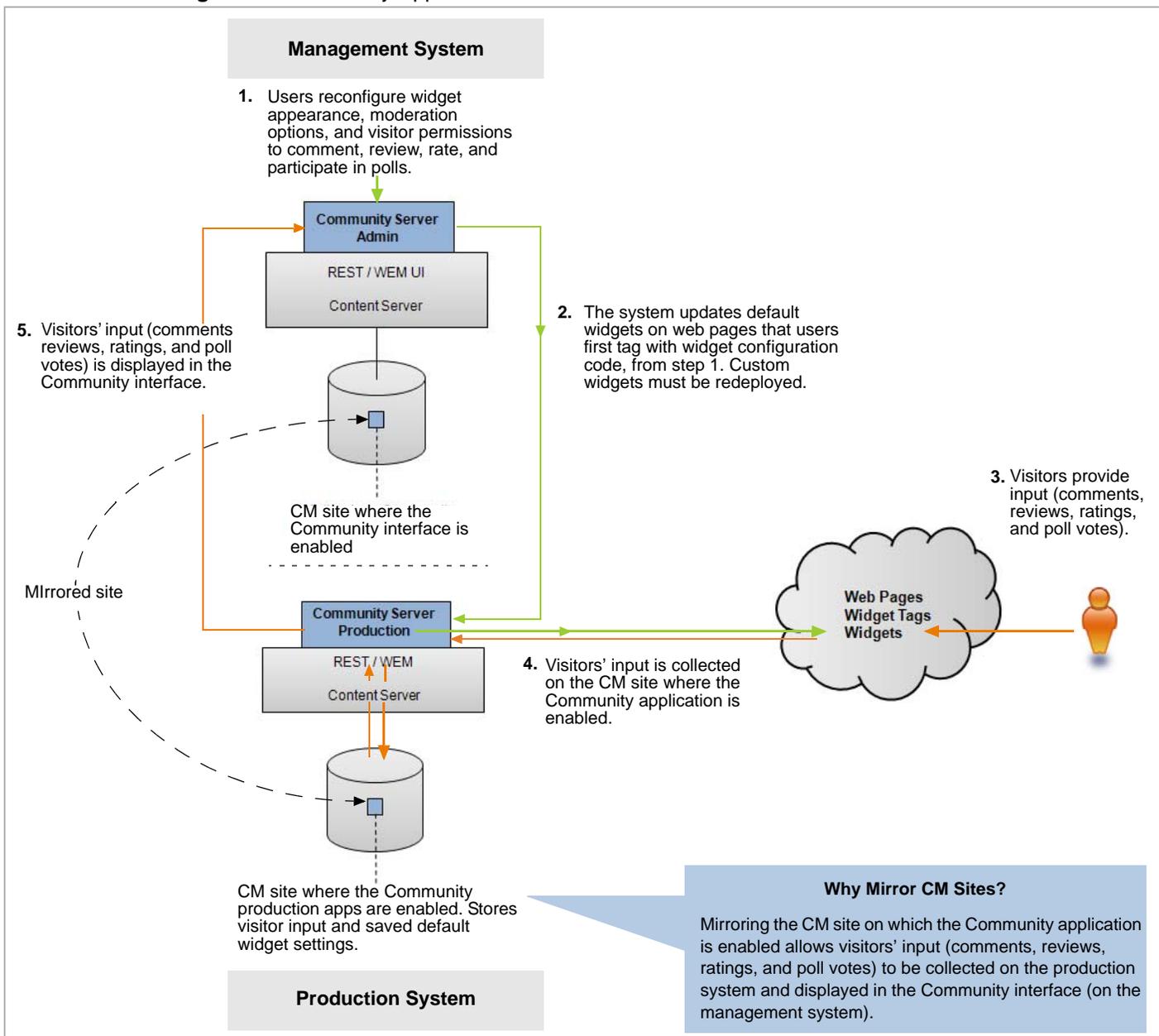
**Table 1:** Community Widget Tags

Widget Type	Widget Tag	For More Information, See ...
<b>Comments</b>	Comments	<a href="#">Chapter 3, “Working with the Comments Widgets” on page 35.</a>
	Comments Summary	
	Links to Topics	
	Recently Commented	
	Most Commented	
<b>Reviews</b>	Reviews	<a href="#">Chapter 4, “Working with the Reviews Widgets” on page 67.</a>
	Reviews Summary	
	Links to Topics	
	Average Rating	
	Top Ranked Topics	
	Recently Reviewed	
	Most Reviewed	
<b>Ratings</b>	Stars Ratings	<a href="#">Chapter 5, “Working with Ratings Widgets” on page 103.</a>
	Thumbs Up/Down Ratings	
	Like It Ratings	
	Recommend Ratings	
	Average Rating	
	Recently Rated	
	Most Rated	
<b>Login Bar</b>	Login Bar	<a href="#">Chapter 6, “Working with the Login Bar Widget” on page 137.</a>
<b>Polls</b>	Poll	<a href="#">Chapter 7, “Working with Polls” on page 143.</a>
	Results	

## Deployment Options

Figure 2 illustrates the events that occur when you reconfigure and deploy community widget tags. Widget tags can be deployed with either default settings (stored on the CM site where the Community production applications are enabled), or custom settings. The tags deployed with default settings are dynamically updated when their default permissions and appearance settings are modified and saved in the Community interface. Because custom settings cannot be saved, configuring a tag with custom settings requires redeploying the tag on the desired web pages.

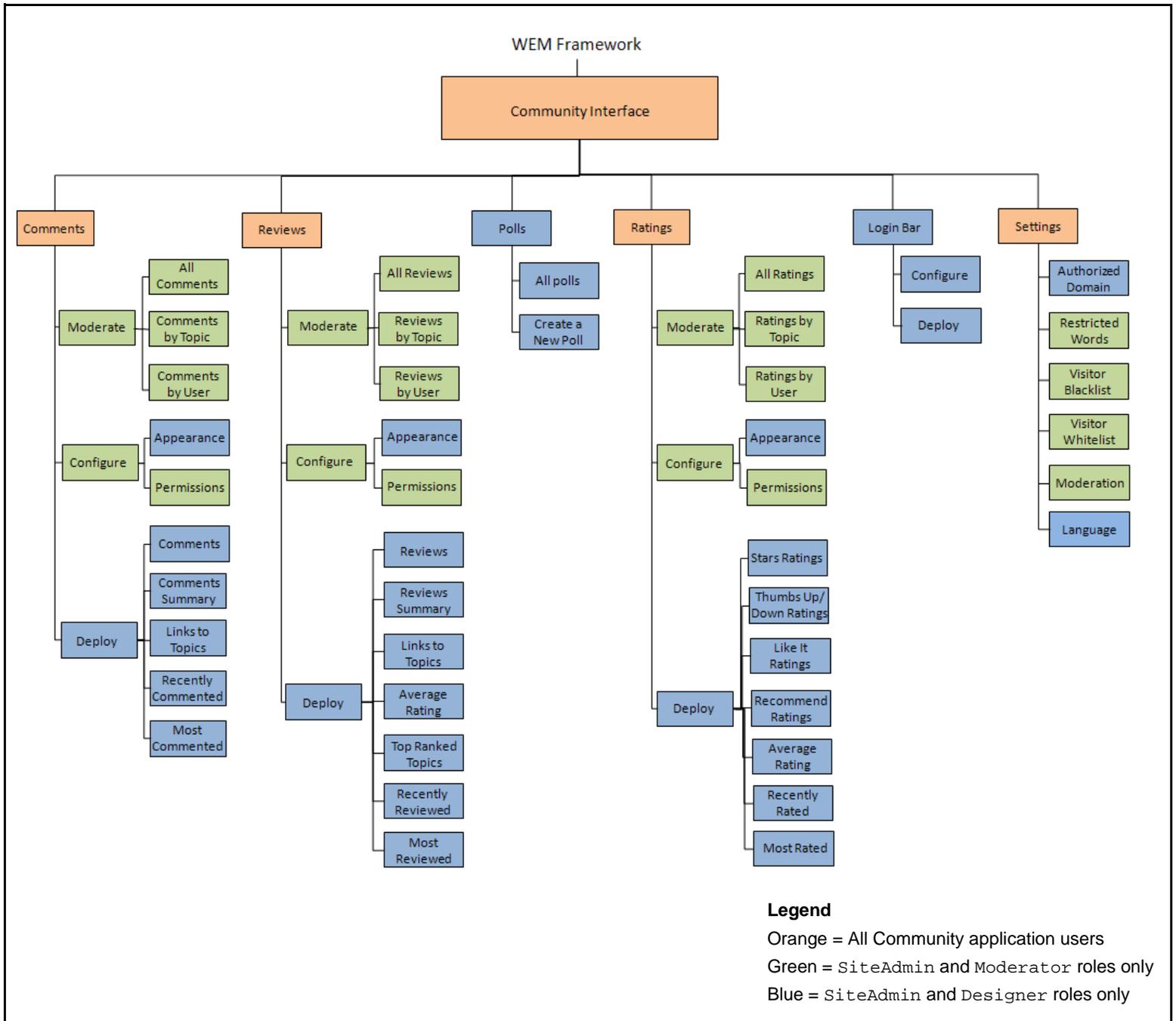
**Figure 2:** Community Application Process Flow



# Community Interface

The Community application provides the Community interface. Figure 3 illustrates the functions associated with the Community interface and identifies the roles that have permissions to those interface functions.

**Figure 3:** Overview of the Community Interface

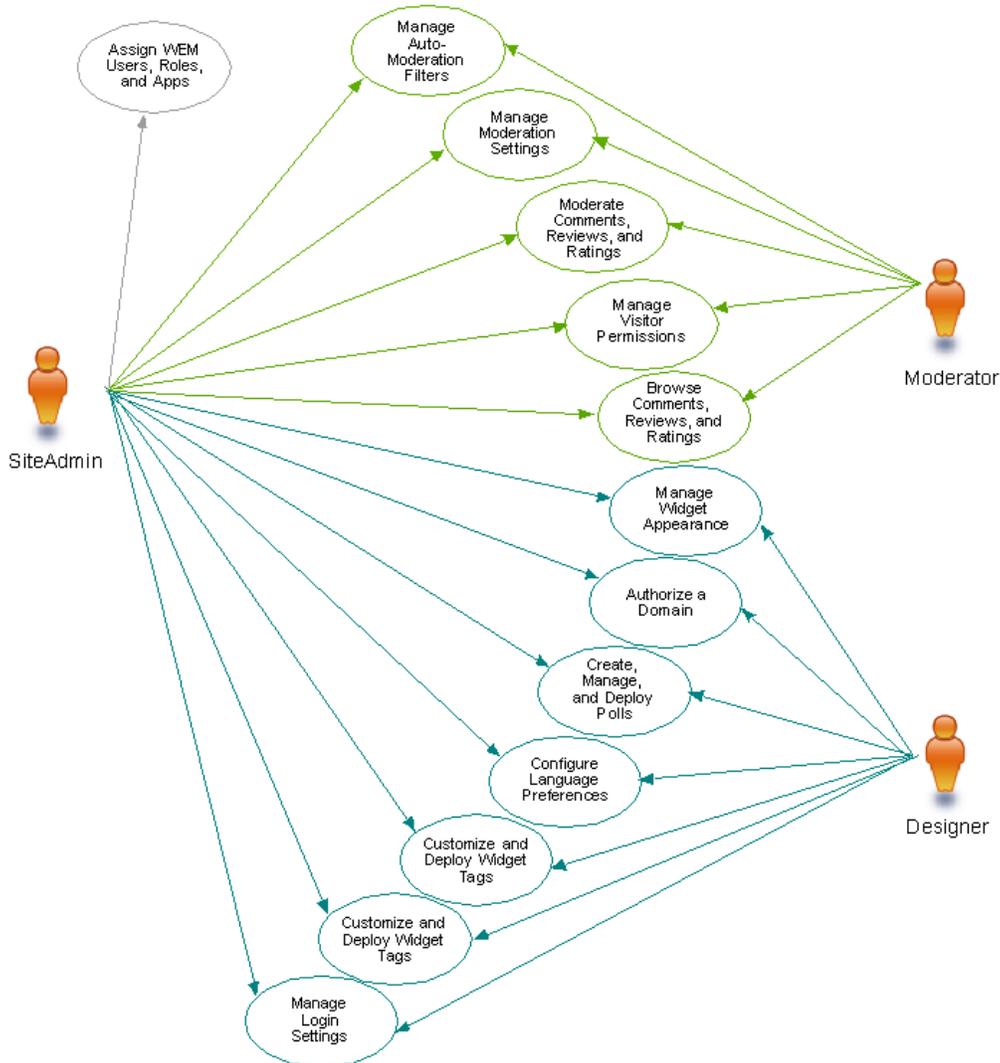


## Community Application Roles and Permissions

Once the Community application is installed, general administrators can use the WEM Admin interface to assign the Community application to the desired site via the roles SiteAdmin, Moderator, and Designer. These roles regulate access to the Community application's interface functions. To authorize the site's users to work with the Community interface, the general administrator must assign each user at least one of the roles assigned to the Community application. For detailed instructions, see the *Oracle WebCenter Sites Administrator's Guide for the Web Experience Management Framework*.

Certain permissions are shared by roles. In [Figure 4](#), the color coded arrows indicate which roles share which permissions. For example, the permissions the SiteAdmin and Moderator roles share are indicated by the green arrows (for example, the permission to browse comments, reviews, and ratings). For a detailed look at the permissions associated with each role, and the permissions site visitors can be assigned, see [Table 2](#), on page 20 and [Table 3](#), on page 22.

**Figure 4:** Community Application Roles and Permissions



When the Community application is installed, the roles `Moderator` and `Designer` are automatically created if they do not already exist.

### **SiteAdmin**

The `SiteAdmin` role is a default WebCenter Sites role. Users assigned the `SiteAdmin` role have administrative permissions to an entire site and its applications, including the Community application. These users have permissions to all Community interface functions, such as configuring the appearance of community widgets, setting visitor permissions to community widgets, managing general community widget settings, and deploying community widget tags on various web pages.

### **Moderator**

Users assigned the `Moderator` role can set visitor permissions to community widgets, configure moderation settings for community widget tags, and enable (or disable) the auto-moderation filters that are used to assist in the moderation process for visitors' comments, reviews, and ratings. These users can also manually moderate comments, reviews, and ratings that are posted to the pages on which community widget tags are deployed.

### **Designer**

Users assigned the `Designer` role can configure the default appearance of community widgets, create polls, and modify a given community widget tag with custom values. These users can also specify the domain on which community widget tags can be deployed, configure the language in which deployed community widgets will be displayed, and use the WebCenter Sites Admin interface to deploy community widget tags and publish the templates.

Table 2 provides an overview of the permissions users can be granted to Community interface functions.

**Table 2:** Community Application Roles and Permissions

Permissions	Site Admin	Moderator	Designer	For More Information, See ...
Manage Visitor Permissions:				
Commenter	✓	✓		<a href="#">“Comments &gt; Configure &gt; Permissions,”</a> on page 51
Reviewer	✓	✓		<a href="#">“Reviews &gt; Configure &gt; Permissions,”</a> on page 83
Rater	✓	✓		<a href="#">“Ratings &gt; Configure &gt; Permissions,”</a> on page 115
Manage Widget Appearance:				
Comments Widget	✓		✓	<a href="#">“Comments &gt; Configure &gt; Appearance,”</a> on page 47
Reviews Widget	✓		✓	<a href="#">“Reviews &gt; Configure &gt; Appearance,”</a> on page 79
Ratings Widgets	✓		✓	<a href="#">“Ratings &gt; Configure &gt; Appearance,”</a> on page 113
Create, Manage, and Deploy Polls	✓		✓	Chapter 7, <a href="#">“Working with Polls”</a>
Configure and Deploy the Login Bar Widget Tag	✓		✓	Chapter 6, <a href="#">“Working with the Login Bar Widget”</a>
Authorize a Domain	✓		✓	<a href="#">“Settings &gt; Authorized Domain,”</a> on page 152
Manage Auto-Moderation Filters:				
Restricted Words	✓	✓		<a href="#">“Settings &gt; Restricted Words,”</a> on page 154
Visitor Blacklist	✓	✓		<a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155
Visitor Whitelist	✓	✓		<a href="#">“Settings &gt; Visitor Whitelist,”</a> on page 157
Manage Moderation Settings	✓	✓		<a href="#">“Settings &gt; Moderation,”</a> on page 158
Configure a Language Preference for Deployed Community Widgets	✓		✓	<a href="#">“Settings &gt; Language,”</a> on page 160
Work with the Cache Tool	✓			Chapter 9, <a href="#">“Working with the Cache Tool”</a>

**Table 2:** Community Application Roles and Permissions *(continued)*

Permissions	Site Admin	Moderator	Designer	For More Information, See ...
Deploy Default Widget Tags:				
Comments	✓		✓	<a href="#">“Deploying a Default Comments Widget Tag,” on page 53</a>
Reviews	✓		✓	<a href="#">“Deploying a Default Reviews Widget Tag,” on page 85</a>
Ratings	✓		✓	<a href="#">“Deploying a Default Ratings Widget Tag,” on page 117</a>
Customize and Deploy Widget Tags:				
Comments	✓		✓	<a href="#">“Deploying Custom Comments Widget Tags,” on page 56</a>
Reviews	✓		✓	<a href="#">“Deploying Custom Reviews Widget Tags,” on page 89</a>
Ratings	✓		✓	<a href="#">“Deploying Custom Ratings Widget Tags,” on page 120</a>
Enable SEO Support for Comments and Reviews	✓		✓	<a href="#">Appendix B, “Enabling SEO Support for Community Widgets”</a>
Deploy the CSS Tag	✓		✓	<a href="#">Appendix C, “Deploying the CSS Tag”</a>
Moderate:				
Comments	✓	✓		<a href="#">“Moderating Comments,” on page 36</a>
Reviews	✓	✓		<a href="#">“Moderating Reviews,” on page 68</a>
Ratings	✓	✓		<a href="#">“Moderating Ratings,” on page 104</a>
Poll Results	✓		✓	<a href="#">“Working with the ‘All Polls’ Screen,” on page 144</a>

Table 3 provides an overview of the permissions site visitors can be granted on the website.

**Table 3:** Visitor Permissions to Community Widgets on the Website

Permissions	Authenticated User	Anonymous User	For More Information, See ...
Log in to the website	✓		“Logging in to the Site as an Authenticated Visitor,” on page 173
Post Comments as an Authenticated Visitor	✓		“Posting Comments as an Authenticated Visitor,” on page 174
Post Reviews as an Authenticated Visitor	✓		“Posting Reviews as an Authenticated Visitor,” on page 176
Modify Approved Comments	✓		“Working with Comments as an Authenticated Visitor,” on page 177
Delete Approved Comments	✓		
Modify Approved Reviews	✓		“Working with Reviews as an Authenticated Visitor,” on page 179
Delete Approved Reviews	✓		
Post Comments and Reviews as a Guest	✓	✓	“Working with Comments and Reviews Widgets as an Anonymous Visitor,” on page 181
Flag Inappropriate Comments and Reviews	✓	✓	“Flagging Inappropriate Posts,” on page 184
Reply to Approved Comments	✓	✓	“Replying to Posted Comments,” on page 185
View a topic’s Review “Detail” Menu	✓	✓	“Viewing Review Details,” on page 186
Rank Individual Comments and Reviews	✓	✓	“Ranking Visitor Comments and Reviews,” on page 187
Share Comments and Reviews to your Facebook or Twitter News Feed	✓	✓	“Sharing Comments and Reviews with Facebook and Twitter,” on page 187
Create a Permalink for a Comment or Review	✓	✓	“Creating and Sharing a Permalink,” on page 188
Submit a Rating for a Topic on the Website	✓	✓	“Working with Ratings Widgets,” on page 189

**Table 3:** Visitor Permissions to Community Widgets on the Website *(continued)*

Permissions	Authenticated User	Anonymous User	For More Information, See ...
Submit a Vote for a Poll	✓	✓	<a href="#">“Working with Poll Widgets,” on page 190</a>



## Chapter 2

# Getting Started

This chapter provides instructions on logging in to the Community interface. The last section of this chapter provides a deployment scenario to help administrators and designers become familiar with the process of deploying community widget tags.

This chapter contains the following sections:

- [Logging In](#)
- [How Do I Deploy Community Widget Tags?](#)
- [Next Steps](#)

## Logging In

The Community application is associated with WebCenter Sites through the WEM Framework. By logging in to a site and accessing the Community application you allow the system to recognize your status as an administrator, moderator, or designer and enable your permissions to Community interface functions.

### To log in to the Community interface

1. Go to the following URL:

`http://<server>:<port>/<context>/login`

where <server> is the host name or IP address of the server running WebCenter Sites, <port> is the port number of the WebCenter Sites application, and <context> is the name of the WebCenter Sites application that was deployed on the server.

2. Access the Community application by logging in to WebCenter Sites. **Log in credentials are case-sensitive.**

In this guide we are using the default login credentials of a general administrator:

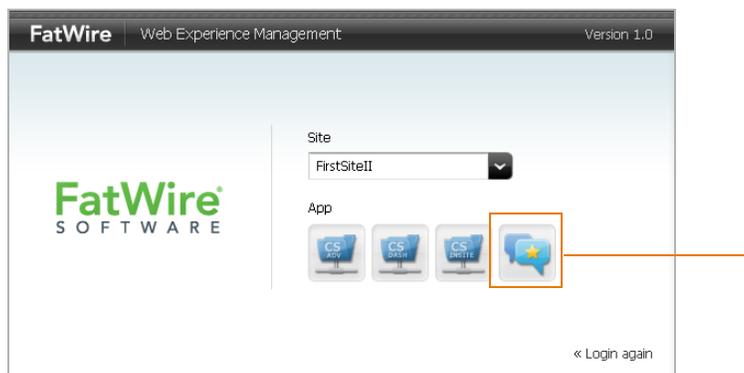
**Username:** fwadmin

**Password:** xceladmin



The screenshot shows the 'FatWire Web Experience Management Version 1.0' login page. On the left is the FatWire logo. On the right, under the heading 'SECURE USER LOGIN', there are two input fields: 'Username' and 'Password'. Below the password field is a link for 'Forgot password?'. A 'Login' button is centered below the fields. At the bottom right, there is a checkbox labeled 'Remember me'.

3. Click **Login**.
4. If you are logging in for the first time, the following screen is rendered; otherwise, you are logged in to the application you last visited:



The screenshot shows the 'FatWire Web Experience Management Version 1.0' application selection screen. On the left is the FatWire logo. On the right, there is a 'Site' dropdown menu with 'FirstSiteII' selected. Below it, there is an 'App' section with four icons: 'CS Adv', 'CS Web', 'CS WEM', and 'Community'. The 'Community' icon, which features a speech bubble with a plus sign, is highlighted with an orange box and has an orange line pointing to it from the right. At the bottom right, there is a link that says '<< Login again'.

Select the site to which the Community application is assigned, and the **Community application** icon.

If you are an administrator or a moderator, the first screen you see is the “All Comments” screen (shown in [Figures 5 and 6](#)).

### Note

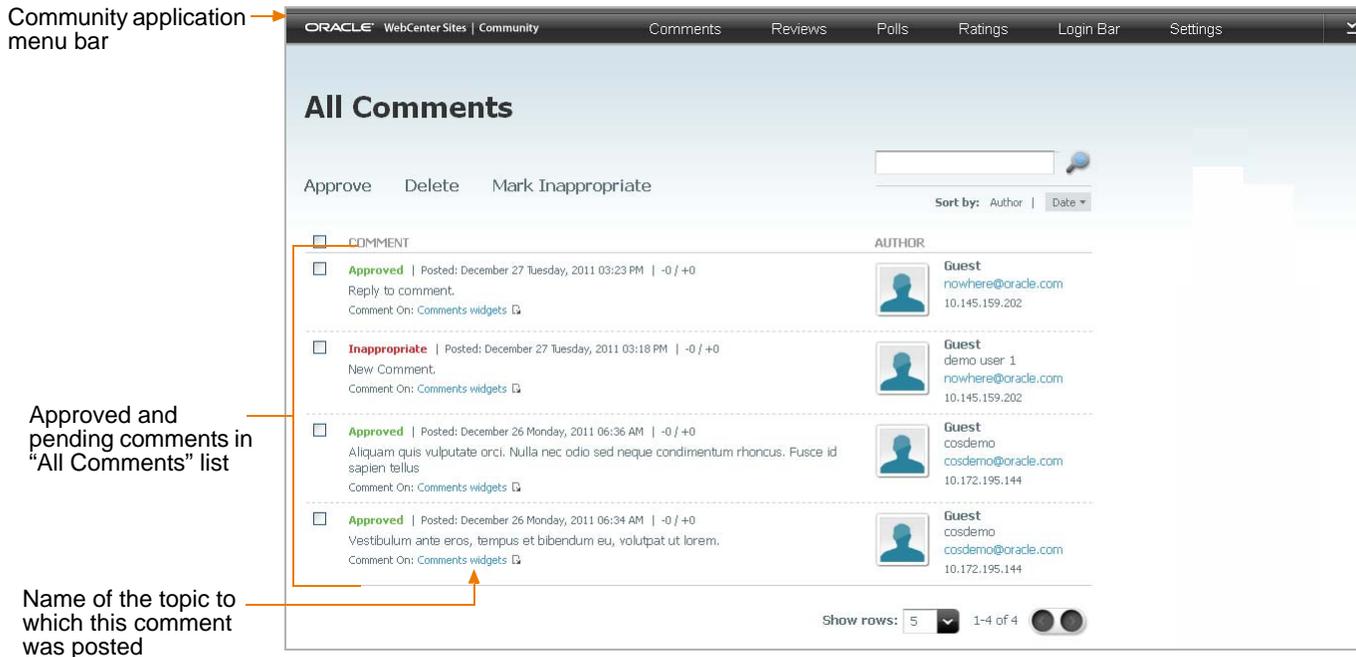
If you are a designer, the first screen you see is the “All Polls” screen. For information about the “All Polls” screen, see “[Working with the ‘All Polls’ Screen,](#)” on page 144.

If you are an administrator or moderator and this is the first time the Community application is being used, then there will be no comments listed in the “All Comments” screen, as shown in [Figure 5](#).

**Figure 5:** Unpopulated Community application “All Comments” screen



If you are an administrator or moderator and the Comments widget tag has already been configured and deployed on a web page, and site visitors have posted comments to that page, you will see those comments in the “All Comments” screen, as shown in [Figure 6](#).

**Figure 6:** Populated Community application “All Comments” screen

Administrators and moderators can also view visitors’ comments, reviews, and ratings by either the topic to which they were posted or by the name of the site visitor who posted them.

## How Do I Deploy Community Widget Tags?

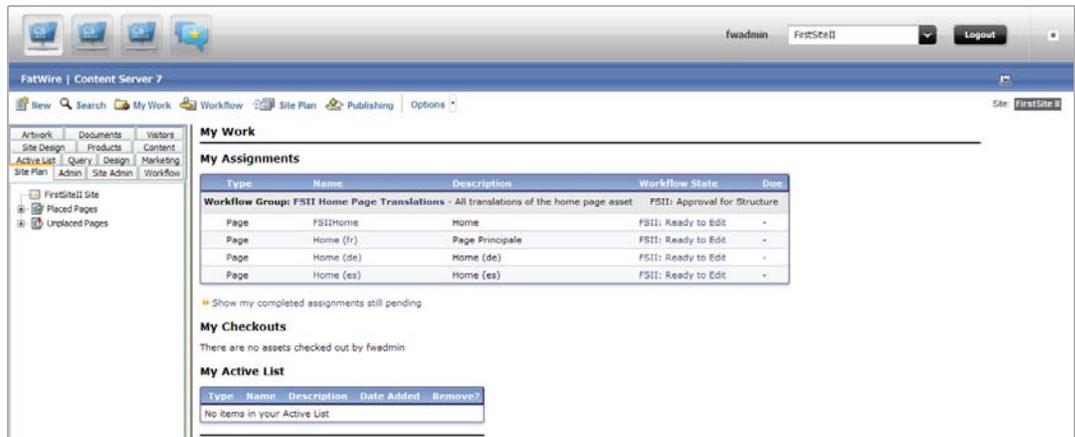
This scenario is for administrators and designers, and provides instructions for deploying a Comments widget tag. In this scenario, the Community application is assigned to the WebCenter Sites `FirstSiteII` sample site. You will be inserting the Comments widget tag into the `Content_C/FSIIDetail` template. This template renders `Content_C` assets on the website. Once this template is published, any web pages that display the content of a `Content_C` asset (for example, the content of the `FSIIAbout` asset) will also display a Comments widget.

### Note

- To deploy a widget tag, you must have access to the WebCenter Sites Admin interface (**Admin**). Ensure that the WebCenter Sites Admin application is assigned to the same site as the Community application.
- If you wish to reconfigure the default settings and visitor permissions for the Comments widget tag before you deploy the tag, see [“Reconfiguring the Comments Widget’s Default Settings,”](#) on page 47.
- If you wish to test the current widget tag settings and visitor permissions without using a WebCenter Sites CM site, insert the tag into an `html` page instead to view the configurations you are starting with.



- 2) In the applications bar, click the WebCenter Sites **Admin** icon to open the WebCenter Sites Admin interface:



- c. In the WebCenter Sites Admin interface, locate the template into which you wish to insert the widget tag (Content\_C/FSIIDetail template in this example):
- 1) From the start menu options, click **Search**.
  - 2) In the “Search” results list, select **Find Template**.
  - 3) In the “Search for Templates” form, click **Search**.
  - 4) In the “List of Templates” screen, select the template into which you wish to insert the widget tag. (Content\_C/FSIIDetail template in this example.)
- d. In the template’s “Inspect” form, click **Edit**.
- e. In the template’s “Element” screen, insert (**Ctrl+V**) the widget tag into the “Element Logic” field in the exact location you want to display the widget.

The screenshot shows the 'Template: FSIIDetail' 'Element' screen. The 'Element Logic' field contains the following code:

```

assetType='<%=cos.GetVar("c")%>'
editor='fckeditor!/'>
</div>

<div id="comments_container"></div>
<script type="text/javascript">
if(cos == undefined)
{
var cos = {};
cos.widgets = new Array();
}
else if(cos.widgets == undefined)
{
cos.widgets = new Array();
}
cos.widgets.push({name:"comments", version:"0.1",
</script>
<script type="text/javascript" src="http://coserv
</div>

</cs:ftcs>

```

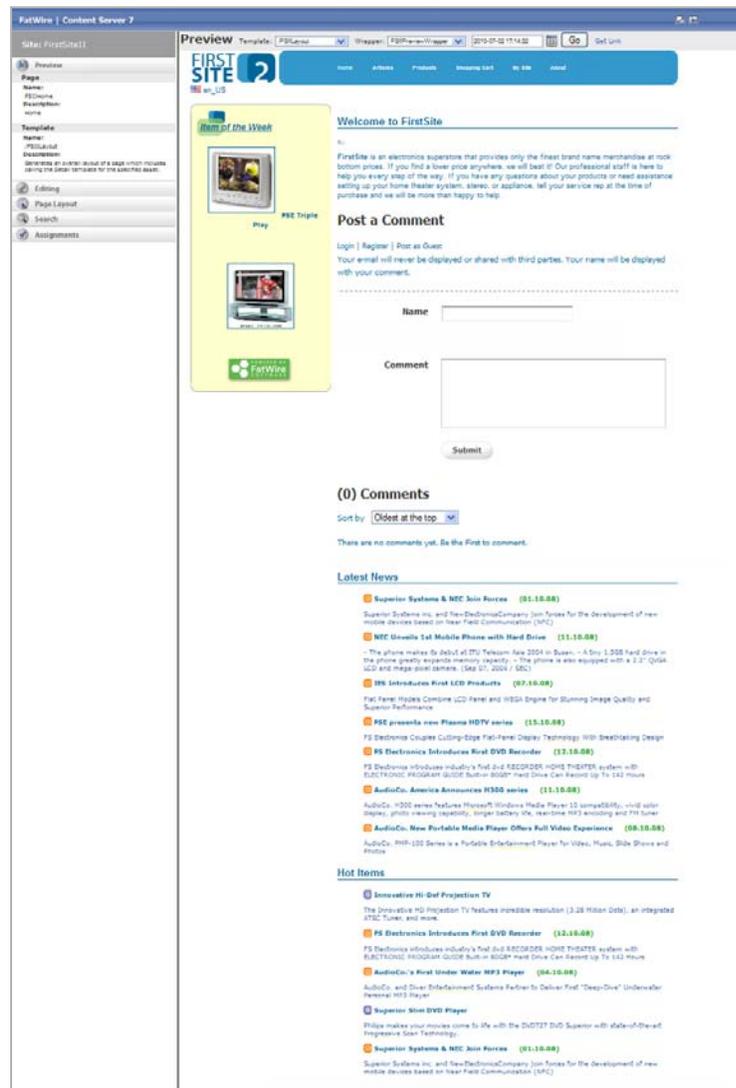
An orange box highlights the widget tag insertion point in the 'Element Logic' field.

Comments widget tag inserted into the “Element Logic” field.

- f. Click **Save Changes**.

5. If you are deploying multiple widgets on the same page, deploy the CSS tag. The CSS tag loads all widget CSS files at the same time, decreasing the load time of a page on which multiple widgets are deployed. For detailed information and instructions about deploying the CSS tag, see [Appendix C, “Deploying the CSS Tag.”](#)
6. Preview the widget to ensure it is properly rendered on the website:
  - a. Locate the asset that is rendered by the template containing the widget tag. (For instructions on searching, see [step c.](#))
  - b. In the asset’s “Inspect” form, click **Preview.**

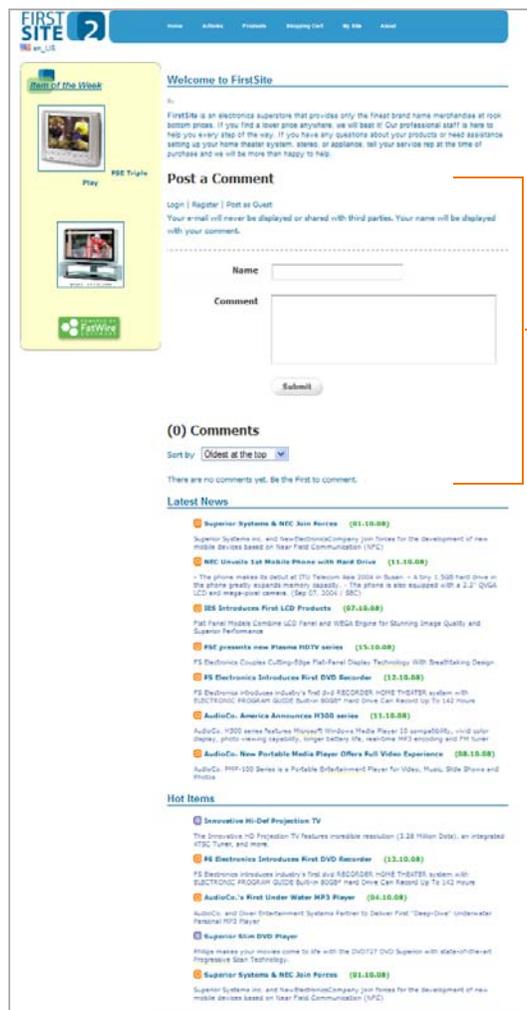
The asset’s preview opens displaying the widget (Comments widget in this example):



- If you want to publish the template to the website, see [step 7.](#)
- If you want to reconfigure default settings for the Comments widget tag, see [“Reconfiguring the Comments Widget’s Default Settings,”](#) on page 47.

- If you want to reconfigure and deploy another type of widget tag, see one of the following chapters:
    - Chapter 4, “Working with the Reviews Widgets”
    - Chapter 5, “Working with Ratings Widgets”
    - Chapter 6, “Working with the Login Bar Widget”
    - Chapter 7, “Working with Polls”
  - If you want to reconfigure general community widget settings, such as security, language, and moderation settings, see Chapter 8, “Reconfiguring General Community Widget Settings.”
7. Publish the template. For instructions, see the *Oracle WebCenter Sites Administrator’s Guide*.
  8. Access the website to view the widget on a web page.

The web page should look similar to the following:



Web page displaying the Comments widget.

## Next Steps

The next chapters provide information and instructions for moderating visitors' comments, reviews, and ratings, as well as viewing poll results. The next chapters also provides information about configuring and deploying the Comments, Reviews, ratings, polls, and Login Bar widgets. You will also learn how to reconfigure general community widget settings, such as the language in which community widgets are displayed on the website.



## Chapter 3

# Working with the Comments Widgets

This chapter provides a quick reference for moderating the comments posted to your site by visitors, configuring the visitor permissions and appearance settings for the Comments widget, and deploying the Comments widget (and all related widgets) to a web page with either dynamic (default) or custom settings.

This chapter contains the following sections:

- [Overview](#)
- [Moderating Comments](#)
- [Reconfiguring the Comments Widget's Default Settings](#)
- [Deploying the Comments Widget Tags](#)

## Overview



The **Comments** menu is used to work with the Comments widget and all of its related widgets (Comments Summary, Links to Topics, Recently Commented, and Most Commented). The “Comments” menu provides you with options to moderate the comments that visitors post to the pages of your website, configure visitor permissions and appearance settings for the Comments widget, and deploy the Comments widget and all related widgets (with either default or custom settings) to web pages.

The instructions in this section are written as a quick reference and it is assumed you are already logged in to the Community interface. If you need instructions for logging in, see [“Logging In,” on page 26](#).

## Moderating Comments

This section is for administrators and moderators. You can access and moderate the comments that site visitors post to your website in the following ways:

### Comments > Moderate > All Comments

This screen is used to moderate all of the comments that have been posted to the pages of your website. The comments are listed in the order they were posted (most recent to least recent). You can sort the comments in this list by either the visitor who posted the comment or the date the comment was posted.

COMMENT	AUTHOR
<input type="checkbox"/> <b>Approved</b>   Posted: December 27 Tuesday, 2011 03:23 PM   -0 / +0 Reply to comment. Comment On: <a href="#">Comments widgets</a>	<b>Guest</b> nowhere@oracle.com 10.145.159.202
<input type="checkbox"/> <b>Inappropriate</b>   Posted: December 27 Tuesday, 2011 03:18 PM   -0 / +0 New Comment. Comment On: <a href="#">Comments widgets</a>	<b>Guest</b> demo user 1 nowhere@oracle.com 10.145.159.202
<input type="checkbox"/> <b>Approved</b>   Posted: December 26 Monday, 2011 06:36 AM   -0 / +0 Aliquam quis vulputate orci. Nulla nec odio sed neque condimentum rhoncus. Fusce id sapien tellus Comment On: <a href="#">Comments widgets</a>	<b>Guest</b> cosdemo cosdemo@oracle.com 10.172.195.144
<input type="checkbox"/> <b>Approved</b>   Posted: December 26 Monday, 2011 06:34 AM   -0 / +0 Vestibulum ante eros, tempus et bibendum eu, volutpat ut lorem. Comment On: <a href="#">Comments widgets</a>	<b>Guest</b> cosdemo cosdemo@oracle.com 10.172.195.144

**Table 4:** Moderating All Comments

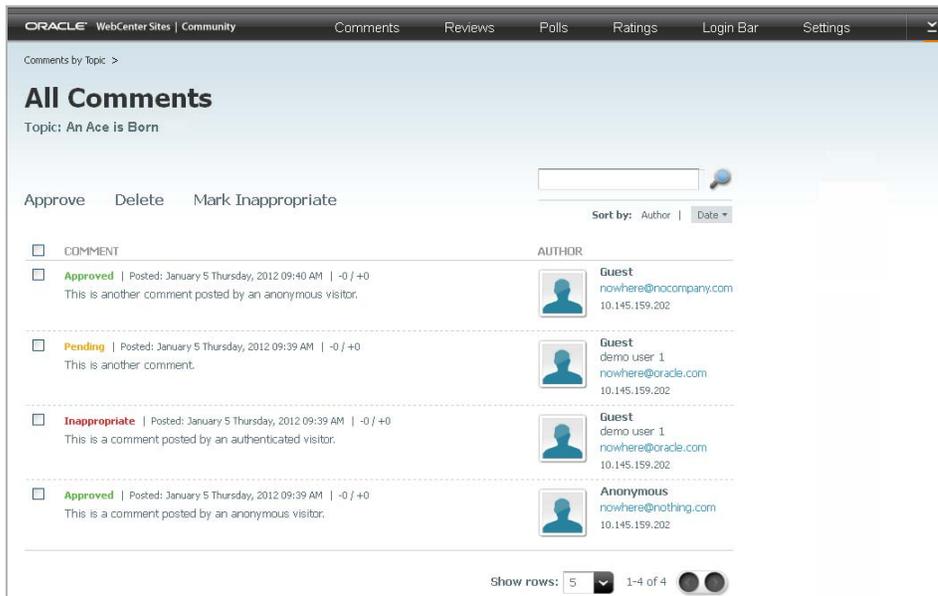
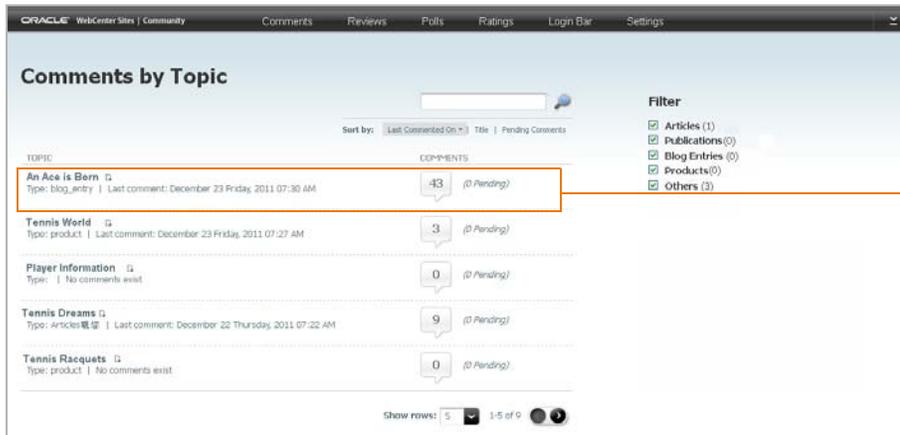
Action	Instructions
<b>Approve visitors' comments</b>	<ol style="list-style-type: none"> <li>Point to the comment you wish to approve.</li> <li>In the menu that appears, select <b>Approve</b>. The approved comment is displayed on the website</li> </ol>
<b>Delete visitors' comments</b>	<ol style="list-style-type: none"> <li>Point to the comment you wish to delete.</li> <li>In the menu that appears, select <b>Delete</b>.</li> <li>In the confirmation box, click <b>OK</b>. The comment is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a comment as inappropriate</b>	<ol style="list-style-type: none"> <li>Point to the comment that contains inappropriate or inaccurate content.</li> <li>In the menu that appears, select <b>Mark Inappropriate</b>. The status of the comment changes to <b>Inappropriate</b>. Marking the comment as inappropriate indicates to other moderators that the comment must be modified before being displayed on the website.</li> </ol> <p><b>Note:</b> Comments can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, "Reconfiguring General Community Widget Settings."</a></p>
<b>Modify a comment</b>	<ol style="list-style-type: none"> <li>Point to the comment you wish to modify.</li> <li>In the menu that appears, click <b>Edit</b>. The "Edit Comment" screen is displayed.</li> <li>In the "Comment" field, make the desired changes to the content.</li> <li>Click <b>Save</b>. The modified comment is displayed in the Community interface and on the web page to which it was posted.</li> </ol> <p><b>Note:</b> If manual moderation is enabled ("<a href="#">Settings &gt; Moderation</a>"), manually approve the comment you modified to display it on the web page.</p>
<b>Add an authenticated visitor to the "Visitor Whitelist"</b>	<ol style="list-style-type: none"> <li>Point to a comment posted by the authenticated visitor you wish to add to the whitelist.</li> <li>In the menu that appears, click <b>Whitelist Author</b>. The "Visitor Whitelist" screen opens, and the visitor's name is displayed in the list of whitelisted visitors.</li> </ol> <p><b>Note:</b> For more information about the "Visitor Whitelist," see "<a href="#">Settings &gt; Visitor Whitelist</a>," on page 157.</p>

**Table 4:** Moderating All Comments (*continued*)

Action	Instructions
<b>Add a visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>1. Point to a comment posted by the visitor you wish to add to the blacklist.</li> <li>2. In the menu that appears, click <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens. If the visitor is authenticated, the visitor’s user name is displayed in the list of blacklisted visitors. If the visitor is anonymous, the visitor’s IP address is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Blacklist,” see <a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155.</p>
<b>Search for a comment containing specific content</b>	<ol style="list-style-type: none"> <li>1. In the “Search” field, enter the criteria you wish to search for.</li> <li>2. Click the <b>magnifying glass</b> () icon. Only comments containing the search criteria you specified are listed in the “All Comments” screen.</li> </ol>
<b>Identify a flagged comment</b>	A comment that is flagged on the website by a visitor will be listed in the “All Comments” screen with <b>Flagged</b> posted next to its <b>Approved</b> status.
<b>View a topic on the website</b>	<ol style="list-style-type: none"> <li>1. In the list of comments, navigate to the comment that is posted to the desired topic.</li> <li>2. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed Comments widget, and the comments that visitors posted to the page.</li> </ol>
<b>View all comments posted by a specific visitor</b>	<ol style="list-style-type: none"> <li>1. In the list of comments, navigate to the comment that is posted by the desired authenticated visitor.</li> <li>2. Click the visitor’s name. The “User Comments” screen opens and displays only the comments posted by the selected visitor.</li> </ol> <p><b>Note:</b> For information about moderating comments by user, see <a href="#">“Comments &gt; Moderate &gt; Comments by User,”</a> on page 43.</p>
<b>View all comments posted to a specific topic</b>	<ol style="list-style-type: none"> <li>1. In the list of comments, navigate to the comment that is posted to the desired topic.</li> <li>2. Click the name of the topic. The “All Comments” screen opens and displays only the comments posted to the selected topic.</li> </ol> <p><b>Note:</b> For information about moderating comments by topic, see <a href="#">“Comments &gt; Moderate &gt; Comments by Topic,”</a> on page 39.</p>

## Comments > Moderate > Comments by Topic

This screen is used to sort all comments that have been posted to the pages of your website by the title of the topic (web page) to which the comments are posted. When you select the title of a topic from this screen, the “All Comments” screen opens and displays only the comments posted to that topic.



Clicking the name of a topic in the “Comments by Topic” screen opens the “All Comments” screen, which displays only the comments posted to that topic.

**Table 5: Moderating Comments by Topic**

Action	Instructions
<b>Filter topics by resource type (topic type)</b>	<ol style="list-style-type: none"> <li data-bbox="651 373 1331 514">1. In the “Filter” panel, select the checkbox(es) of the desired resource type(s). The web pages (topics) with Comments widgets of the selected resource type(s) are the only web pages listed in the “Comments by Topic” screen.</li> <li data-bbox="651 525 1331 640">2. Select the name of the topic whose comments you wish to view. The “All Comments” screen opens and displays only the comments posted to the selected topic.</li> </ol> <p data-bbox="651 651 1331 804"><b>Note:</b> The resource types (topic types) available in the “Filter” panel are dynamically modified by the current category file (“Comments &gt; Configure &gt; Appearance”). “Others” refers to Comments widgets (and topics) that do not belong to a resource type.</p>
<b>Approve visitors’ comments</b>	<ol style="list-style-type: none"> <li data-bbox="651 825 1331 877">1. In the “Comments by Topic” screen, select the topic whose comments you wish to moderate.</li> <li data-bbox="651 888 1331 940">2. In the “All Comments” screen for the topic you selected, point to the comment you wish to approve.</li> <li data-bbox="651 951 1331 1014">3. In the menu that appears, select <b>Approve</b>. The approved comment is displayed on the website</li> </ol>
<b>Delete visitors’ comments</b>	<ol style="list-style-type: none"> <li data-bbox="651 1035 1331 1087">1. In the “Comments by Topic” screen, select the topic whose comments you wish to moderate.</li> <li data-bbox="651 1098 1331 1150">2. In the “All Comments” screen for the topic you selected, point to the comment you wish to delete.</li> <li data-bbox="651 1161 1331 1192">3. In the menu that appears, select <b>Delete</b>.</li> <li data-bbox="651 1203 1331 1287">4. In the confirmation box, click <b>OK</b>. The comment is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a comment as inappropriate</b>	<ol style="list-style-type: none"> <li data-bbox="651 1308 1331 1360">1. In the “Comments by Topic” screen, select the topic whose comments you wish to moderate.</li> <li data-bbox="651 1371 1331 1455">2. In the “All Comments” screen for the topic you selected, point to the comment that contains inappropriate or inaccurate content.</li> <li data-bbox="651 1465 1331 1612">3. In the menu that appears, select <b>Mark Inappropriate</b>. The status of the comment changes to <b>Inappropriate</b>. Marking the comment as inappropriate indicates to other moderators that the comment must be modified before being displayed on the website.</li> </ol> <p data-bbox="651 1623 1331 1730"><b>Note:</b> Comments can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, “Reconfiguring General Community Widget Settings.”</a></p>

**Table 5:** Moderating Comments by Topic *(continued)*

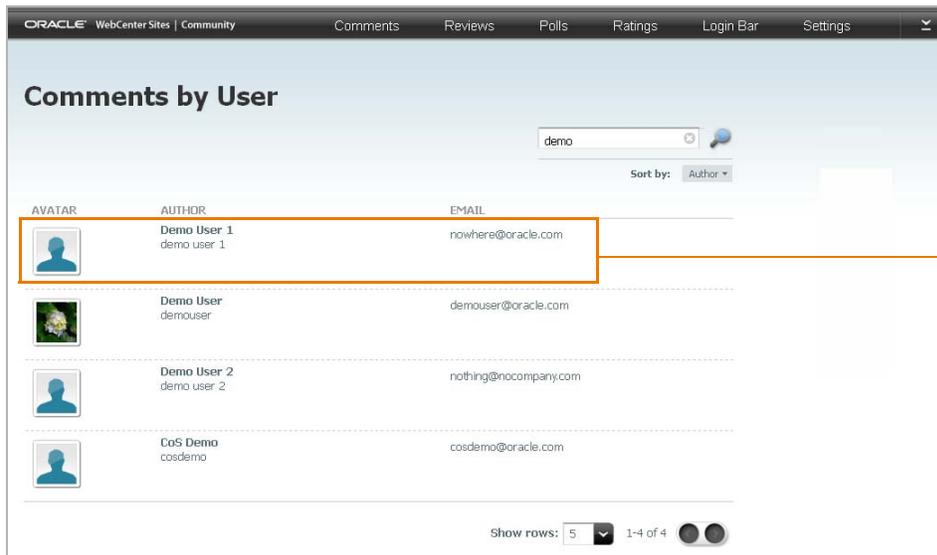
Action	Instructions
<b>Modify a comment</b>	<ol style="list-style-type: none"> <li>In the “Comments by Topic” screen, select the topic whose comments you wish to moderate.</li> <li>In the “All Comments” screen for the topic you selected, point to the comment you wish to modify.</li> <li>In the menu that appears, click <b>Edit</b>. The “Edit Comment” screen is displayed.</li> <li>In the “Comment” field, make the desired changes to the content.</li> <li>Click <b>Save</b>. The modified comment is displayed in the Community interface and on the web page to which it was posted.</li> </ol> <p><b>Note:</b> If manual moderation is enabled (“<a href="#">Settings &gt; Moderation</a>”), manually approve the comment you modified to display it on the web page.</p>
<b>Add an authenticated visitor to the “Visitor Whitelist”</b>	<ol style="list-style-type: none"> <li>In the “Comments by Topic” screen, select the topic whose comments you wish to moderate.</li> <li>In the “All Comments” screen for the topic you selected, point to a comment posted by the authenticated visitor you wish to add to the whitelist.</li> <li>In the menu that appears, click <b>Whitelist Author</b>. The “Visitor Whitelist” screen opens, and the visitor’s name is displayed in the list of whitelisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Whitelist,” see “<a href="#">Settings &gt; Visitor Whitelist</a>,” on page 157.</p>
<b>Add a visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>In the “Comments by Topic” screen, select the topic whose comments you wish to moderate.</li> <li>In the “All Comments” screen for the topic you selected, point to a comment posted by the visitor you wish to add to the blacklist.</li> <li>In the menu that appears, click <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens. If the visitor is authenticated, the visitor’s user name is displayed in the list of blacklisted visitors. If the visitor is anonymous, the visitor’s IP address is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Blacklist,” see “<a href="#">Settings &gt; Visitor Blacklist</a>,” on page 155.</p>
<b>Search for a topic</b>	<ol style="list-style-type: none"> <li>In the “Search” field, enter search criteria identifying the desired topic.</li> <li>Click the <b>magnifying glass</b> () icon. Only topics containing the specified search criteria are listed in the “Comments by Topic” screen.</li> </ol>

**Table 5:** Moderating Comments by Topic *(continued)*

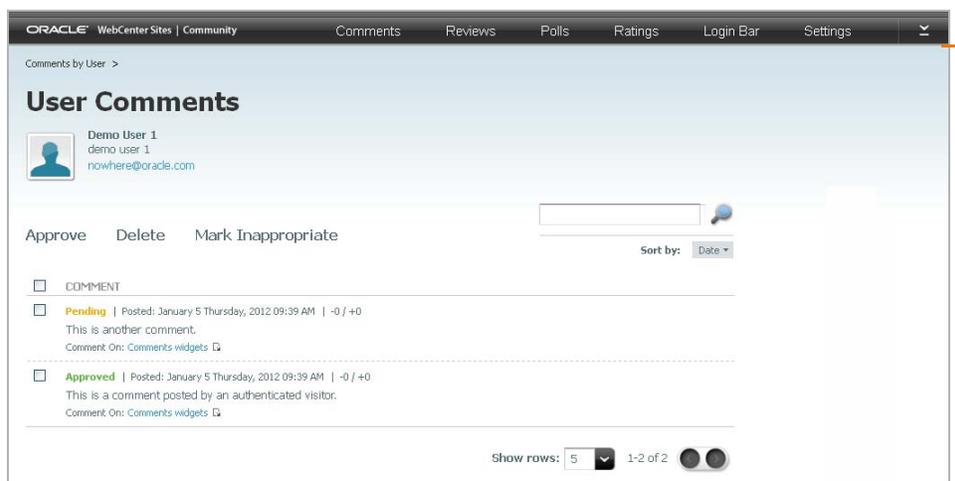
Action	Instructions
<b>Identify a flagged comment</b>	<ol style="list-style-type: none"><li data-bbox="651 373 1326 426">1. In the “Comments by Topic” screen, select the topic whose comments you wish to view.</li><li data-bbox="651 436 1326 489">2. In the “All Comments” screen for the topic you selected, identify any flagged comments. A comment that is flagged on the website will appear in the list of comments with <b>Flagged</b> posted next to its <b>Approved</b> status.</li></ol>
<b>View a topic on the website</b>	<ol style="list-style-type: none"><li data-bbox="651 600 1326 653">1. In the “Comments by Topic” screen, navigate to the desired topic.</li><li data-bbox="651 663 1326 716">2. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed Comments widget, and the comments visitors posted to the page.</li></ol>

## Comments > Moderate > Comments by User

This screen is used to sort all comments posted to the website by the names of the visitors who posted the comments. When you select a visitor's name from this screen, the "User Comments" screen opens and displays the comments the visitor has posted along with the visitor's user name, profile picture, and e-mail address (if available).



Clicking a visitor's user name in the "Comments by User" screen opens the "User Comments" screen, which displays the selected visitor's profile information along with all the comments that visitor has posted to the website.



**Table 6:** Moderating Comments by User

Action	Instructions
<b>Approve visitors' comments</b>	<ol style="list-style-type: none"> <li>1. In the “Comments by User” screen, select the name of the visitor whose comments you wish to moderate.</li> <li>2. In the “User Comments” screen, point to the comment you wish to approve.</li> <li>3. In the menu that appears, select <b>Approve</b>. The approved comment is displayed on the website</li> </ol>
<b>Delete visitors' comments</b>	<ol style="list-style-type: none"> <li>1. In the “Comments by User” screen, select the name of the visitor whose comments you wish to moderate.</li> <li>2. In the “User Comments” screen, point to the comment you wish to delete.</li> <li>3. In the menu that appears, select <b>Delete</b>.</li> <li>4. In the confirmation box, click <b>OK</b>. The comment is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a comment as inappropriate</b>	<ol style="list-style-type: none"> <li>1. In the “Comments by User” screen, select the name of the visitor whose comments you wish to moderate.</li> <li>2. In the “User Comments” screen, point to the comment that contains inappropriate or inaccurate content.</li> <li>3. In the menu that appears, select <b>Mark Inappropriate</b>. The status of the comment changes to <b>Inappropriate</b>. Marking the comment as inappropriate indicates to other moderators that the comment must be modified before being displayed on the website.</li> </ol> <p><b>Note:</b> Comments can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, “Reconfiguring General Community Widget Settings.”</a></p>
<b>Modify a comment</b>	<ol style="list-style-type: none"> <li>1. In the “Comments by User” screen, select the name of the visitor whose comments you wish to moderate.</li> <li>2. In the “User Comments” screen, point to the comment you wish to modify.</li> <li>3. In the menu that appears, select <b>Edit</b>. The “Edit Comment” screen is displayed.</li> <li>4. In the “Comment” field, make the desired changes to the content.</li> <li>5. Click <b>Save</b>. The modified comment is displayed in the Community interface and on the web page to which it was posted.</li> </ol> <p><b>Note:</b> If manual moderation is enabled (“<a href="#">Settings &gt; Moderation</a>”), manually approve the comment you modified to display it on the web page.</p>

**Table 6:** Moderating Comments by User *(continued)*

Action	Instructions
<b>Add an authenticated visitor to the “Visitor Whitelist”</b>	<ol style="list-style-type: none"> <li>In the “Comments by User” screen, do one of the following: <ul style="list-style-type: none"> <li>If you know which authenticated visitor you want to add to the whitelist, point to the visitor’s name. Continue to <a href="#">step 2</a>.</li> <li>If you wish to view the comments posted by the visitor before adding that visitor to the whitelist, do the following: <ol style="list-style-type: none"> <li>Select the name of the visitor whose comments you wish to view.</li> <li>In the “User Comments” screen, point to any of the comments displayed in the list. Continue to <a href="#">step 2</a>.</li> </ol> </li> </ul> </li> <li>In the menu that appears, click <b>Whitelist Author</b>. The “Visitor Whitelist” screen opens, and the visitor’s name is displayed in the list of whitelisted visitors. <b>Note:</b> For more information about the “Visitor Whitelist,” see <a href="#">“Settings &gt; Visitor Whitelist,”</a> on page 157.</li> </ol>
<b>Add an authenticated visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>In the “Comments by User” screen, do one of the following: <ul style="list-style-type: none"> <li>If you know which visitor you want to add to the blacklist, point to the visitor’s name. Continue to <a href="#">step 2</a>.</li> <li>If you wish to view the comments posted by the visitor before adding that visitor to the blacklist, do the following: <ol style="list-style-type: none"> <li>Select the name of the visitor whose comments you wish to view.</li> <li>In the “User Comments” screen, point to any of the comments displayed in the list. (Continue to <a href="#">step 2</a>.)</li> </ol> </li> </ul> </li> <li>In the menu that appears, click <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens, and the visitor’s name is displayed in the list of blacklisted visitors. <b>Note:</b> For more information about the “Visitor Blacklist,” see <a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155.</li> </ol>
<b>Search for a specific visitor</b>	<ol style="list-style-type: none"> <li>In the “Search” field, enter the name of the visitor you wish to search for.</li> <li>Click the <b>magnifying glass</b> () icon. Only visitor names that match the specified search criteria are displayed in the “Comments by User” screen.</li> </ol>
<b>Identify a flagged comment</b>	<ol style="list-style-type: none"> <li>In the “Comments by User” screen, select the name of the visitor whose comments you wish to view.</li> <li>In the “User Comments” screen, identify the flagged comments. A comment that is flagged on the website will appear in the list of comments with <b>Flagged</b> posted next to its <b>Approved</b> status.</li> </ol>

**Table 6:** Moderating Comments by User *(continued)*

Action	Instructions
<b>View a topic on the website</b>	<ol style="list-style-type: none"><li data-bbox="654 369 1323 426">1. In the “Comments by User” screen, select the name of the visitor whose comment is posted to the desired topic.</li><li data-bbox="654 432 1323 489">2. In the “User Comments” screen, navigate to the desired comment.</li><li data-bbox="654 495 1323 619">3. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed Comments widget, and the comments visitors posted to the page.</li></ol>

## Reconfiguring the Comments Widget's Default Settings

Default settings are a widget's saved appearance and permission settings. Administrators and designers can modify the default appearance of the Comments widget by selecting **Comments > Configure > Appearance** in the menu bar. Administrators and moderators can reconfigure the default commenting permissions by selecting **Comments > Configure > Permissions** in the menu bar. Once the permission and appearance settings are saved, the system dynamically updates the Comments widget's default tag in the Community interface and on pages where tags with default settings are deployed.

You can reconfigure the default settings of the Comments widget in the following ways:

### Comments > Configure > Appearance

This screen can be used by administrators and designers to reconfigure the Comments widget's color schema, sort order, and other display properties.

The screenshot shows the 'Comments Appearance' configuration page in Oracle WebCenter Sites. The page is divided into several sections:

- GENERAL:** Includes settings for 'Show Comments' (Display), 'Comment Type' (Flat), 'Pagination' (Yes (40 comments per page)), 'Display Order' (Oldest at the top), 'Comment Submit Box' (At the bottom), 'Comment Ranking' (Enabled for registered visitors), 'Skin' (Default), 'Upload Custom CSS' (with a 'Browse...' button and 'Download the current CSS' link), and 'Login Bar Required' (Yes).
- COMMENTING AUTO DISABLE:** Includes 'Max Number of Comments Allowed' (None) and 'Accept Comments Until' (None), both with 'what is this?' links.
- COMMENT BODY:** Includes 'Max Length' (900 characters) and 'Type' (HTML and media with blocked JS).
- HELP AND DISCLAIMER:** Includes 'Show Help Text' (Display) and 'Show Disclaimer Text' (Display), both with 'what is this?' links. Below each dropdown is a text area containing placeholder text.
- TOPIC CATEGORIES:** Includes 'Upload Categories' (with a 'Browse...' button and 'Download the current category file' link).

A 'Save' button is located at the bottom of the page.

**Table 7:** Reconfiguring the Comments Widget's Appearance Settings

Section	Field	Description
General	Show Comments	Select either <b>Display</b> or <b>Do not Display</b> to specify whether approved comments are displayed to site visitors.
	Comment Type	Select whether your deployment supports a <b>Flat</b> commenting structure, or a <b>Threaded</b> commenting structure which allows replies to comments.
	Pagination	Select the number of approved comments that will be displayed on a given page at one time.
	Display Order	Select one of the following to specify the order in which approved comments will be displayed on a given page: <ul style="list-style-type: none"> <li>• <b>Oldest at the top</b> – Displays the oldest comments at the top of the list of comments.</li> <li>• <b>Newest at the top</b> – Displays the most recently posted comments at the top of the list of comments.</li> <li>• <b>Most ranked at the top</b> – Displays the comments that site visitors frequently rank at the top of the list of comments.</li> <li>• <b>Least ranked at the top</b> – Displays the comments that site visitors rank the least (or have not ranked) at the top of the list of comments.</li> </ul>
	Comment Submit Box	Specify whether the comment field will be displayed <b>At the top</b> or <b>At the bottom</b> of the list of comments on a given page.
	Comment Ranking	Specify whether visitors can rank individual comments as <b>Helpful</b> or <b>Not Helpful</b> . Select one of the following: <ul style="list-style-type: none"> <li>• <b>Enabled for all visitors</b> – All visitors (authenticated and anonymous) can rank the comments posted to a web page.</li> <li>• <b>Enabled for registered visitors</b> – Only authenticated visitors can rank the comments posted to a web page.</li> <li>• <b>Disabled</b> – Individual comments cannot be ranked.</li> </ul>
	Skin	Select <b>Default</b> to use the default color schema for the Comments widget, or select <b>Custom</b> to upload your own CSS file.
	Upload Custom CSS	If you selected the <b>Custom</b> option in the “Skin” field, use the <b>Browse</b> button to select a cascading style sheet (CSS) that will control the appearance of the Comments widget.

**Table 7:** Reconfiguring the Comments Widget's Appearance Settings *(continued)*

Section	Field	Description
<b>General</b> <i>(continued)</i>	Login Bar Required	Select either <b>Yes</b> or <b>No</b> to specify whether a Login Bar widget tag will be embedded in and deployed with the Comments widget tag.  <b>Note:</b> If you select <b>No</b> , you can still deploy a standalone Login Bar widget tag on the same page as the Comments widget (for instructions, see <a href="#">“Deploying a Login Bar Widget Tag,”</a> on page 140).
<b>Commenting Auto Disable</b>	Max Number of Comments Allowed	Enter the maximum number of comments that can be posted to a given web page. Once the maximum number of comments is reached, visitors cannot post comments to that particular web page.
	Accept Comments Until	Enter the maximum number of days comments can be posted to a given web page. Once the specified number of days is reached, visitors cannot post comments to that particular web page.
<b>Comment Body</b>	Max Length	Select either <b>450 characters</b> or <b>900 characters</b> to set the comment field size.
	Type	Specify the type of text that can be entered into the comment field. Select one of the following: <ul style="list-style-type: none"> <li>• <b>Plain text</b> – Visitors can only enter text into the comment field.</li> <li>• <b>HTML with blocked JS</b> – Visitors can enter text and basic HTML tags into the comment field.</li> <li>• <b>HTML and Media with blocked JS</b> – Visitors can enter text, basic HTML tags, videos, and so on into the comment field.</li> </ul>
<b>Help and Disclaimer</b>	Show Help Text	Select either <b>Display</b> or <b>Do not display</b> to specify whether the Comments widget will display help text to visitors.
	Help Text	If you selected <b>Display</b> in the “Show Help Text” field, enter the help text you wish to display to visitors. When the Comments widget is deployed, the help text is displayed above the comments submit box.  <b>Note:</b> This field supports plain text and basic HTML tags. You can use HTML tags to control the appearance of your help text.
	Show Disclaimer Text	Select either <b>Display</b> or <b>Do not display</b> to specify whether the Comments widget will display a disclaimer to visitors.

**Table 7:** Reconfiguring the Comments Widget's Appearance Settings *(continued)*

Section	Field	Description
<b>Help and Disclaimer</b> <i>(continued)</i>	Disclaimer Text	<p>If you selected <b>Display</b> in the “Show Disclaimer Text” field, enter the disclaimer text you wish to display to visitors. When the Comments widget is deployed, the disclaimer is displayed beneath the comments submit box.</p> <p><b>Note:</b> This field supports plain text and basic HTML tags. You can use HTML tags to control the appearance of your disclaimer.</p>
<b>Topic Categories</b>	Upload Categories	<p>Use the <b>Browse</b> button to upload the desired category file, which will dynamically modify the resource types available from:</p> <ul style="list-style-type: none"> <li>• The “Resource Type” field in the Comments widget tag’s custom deployment screen (“<a href="#">Comments &gt; Deploy &gt; Comments &gt; Custom Settings</a>”).</li> <li>• The “Resource Type” field in the Recently Commented (“<a href="#">Comments &gt; Deploy &gt; Recently Commented</a>”) and Most Commented (“<a href="#">Comments &gt; Deploy &gt; Most Commented</a>”) widget tag deployment screens.</li> <li>• The “Comments by Topic” screen’s “Filter” panel (“<a href="#">Comments &gt; Moderate &gt; Comments by Topic</a>”).</li> </ul> <p><b>Note:</b> The category file must be in .txt format, the resource types listed in the file must be in a key=value structure, each resource type listed in the file must be on its own line, and the size of the file you are uploading can be no more than 20MB.</p>

## Comments > Configure > Permissions

This screen can be used by administrators and moderators to display or hide comment fields and set visitors' permissions to comment.

**Table 8:** Reconfiguring Commenting Permissions for Site Visitors

Section	Field
<b>Commenting Enabled</b>	<ul style="list-style-type: none"> <li>• <b>Yes</b> configures the widget to display a comment field.</li> <li>• <b>No</b> hides the comment field, but displays only approved comments, if the widget was previously enabled.</li> </ul>
<b>Who Can Comment?</b>	<p>Available only if the widget is configured to display a comment field:</p> <ul style="list-style-type: none"> <li>• <b>Anonymous users can comment</b> grants commenting permissions to all site visitors. <ul style="list-style-type: none"> <li>Available only if anonymous visitors can comment: <ul style="list-style-type: none"> <li>- <b>User must enter a name</b> requires anonymous visitors to specify a name when posting a comment.</li> <li>- <b>User must enter an email address</b> requires anonymous visitors to specify an email address when posting a comment.</li> <li>- <b>User must enter a captcha</b> requires anonymous visitors to enter the letters and/or numbers of a captcha image before they can post a comment to a web page.</li> </ul> <p><b>Note:</b> By default, this option is greyed out (not available). For information about enabling the “User must enter a captcha” option, see the <i>Oracle WebCenter Sites Developer’s Guide for the Community Application</i>.</p> </li> <li>• <b>Authenticated users only</b> grants commenting permissions only to authenticated site visitors. <ul style="list-style-type: none"> <li>- <b>User can modify and delete their own comment</b> allows authenticated visitors to modify and delete the comments they post.</li> </ul> </li> </ul> </li> </ul>

## Deploying the Comments Widget Tags

This section is for administrators and designers. You can deploy the Comments widget tag and all of its related widget tags with either default or custom settings by selecting **Comments > Deploy > *name of the widget tag you wish to deploy*** in the menu bar. [Table 9](#) lists the widget tags related to commenting functionality and indicates which type of deployment each tag supports.

**Table 9:** Deploying the Comments Widget Tags

Widget Tag	Deploy with Default Settings	Deploy Custom Settings
<b>Comments</b>	✓	✓
<b>Comments Summary</b>	✓ <sup>a</sup>	✓
<b>Links to Topics</b>	✓ <sup>b</sup>	
<b>Recently Commented</b>	✓	✓
<b>Most Commented</b>	✓	✓

- a. If you deploy this widget tag (Comments Summary) on a web page different from the one on which the desired Comments widget is deployed, you will have to link the Comments Summary widget to the Comments widget by specifying a resource ID. For more information, see [“Comments > Deploy > Comments Summary,” on page 61](#).
- b. This widget tag (Links to Topics) must be linked to the web page that displays the desired topic and its comments. For more information, see [“Links to Topics Widget Tag,” on page 215](#).

This section contains the following topics:

- [Deploying a Default Comments Widget Tag](#)
- [Deploying Custom Comments Widget Tags](#)
- [Custom Widget Tag Deployment Screens](#)

## Deploying a Default Comments Widget Tag

Several types of comments widgets can be deployed with default settings: Comments, Comments Summary, Links to Topics, Recently Commented, and Most Commented. The default settings for the Comments widget are its saved settings, which are configured in the “Permissions” screen (**Comments > Configure > Permissions**) and “Appearance” screen (**Comments > Configure > Appearance**). The default settings for all other commenting related widgets are configured out-of-the-box and cannot be reconfigured in the Community interface. Each type of comments widget has its own deployment screen, which displays the widget tag. The default settings are used as property values in the widget tag.

### To deploy default comments widget tags

1. In the menu bar, select **Comments > Deploy > *name of the widget tag you wish to deploy***:
  - **Comments** – to deploy the Comments widget tag. If commenting is enabled (**Comments > Configure > Permissions**), the Comments widget displays a comment field. If displaying comments is enabled (**Comments > Configure > Appearance**), then a list of site visitors’ comments is displayed on the web page.

#### Note

If any of the following deployment scenarios are true, you will have to customize the Comments widget by specifying a resource ID:

- You wish to link the Links to Topics widget to a Comments widget.
- You want to deploy the Comments Summary widget on a web page different from the one on which a Comments widget is deployed.

For more information, see the description of the [Resource ID](#) field in [Table 10, on page 59](#).

- **Comments Summary** – to deploy the Comments Summary widget tag. When deployed on a web page displaying a Comments widget, the Comments Summary widget displays short, summarized information about the number of comments posted to the web page (topic).

#### Note

If you deploy this widget (Comments Summary) on a web page different from the one on which the desired Comments widget is deployed, you will have to link the Comments Summary widget to the Comments widget by specifying a resource ID. For more information, see the description of the [Resource ID](#) field in [Table 11, on page 61](#).

- **Links to Topics** – to deploy the Links to Topics widget tag. When deployed, this widget displays the number of comments posted to a given web page (topic) along with a link to that topic’s comments.

#### Note

The Links to Topics widget tag must be linked to the web page that displays the desired topic and its comments. The link must include the resource ID of the desired topic’s associated Comments widget. For more information, see [line 1](#) in the section “[Links to Topics Widget Tag](#),” on [page 200](#).

- **Recently Commented** – to deploy the Recently Commented widget tag. When deployed, this widget displays a list of the top ten web pages (topics) to which site visitors have recently posted comments.

#### Note

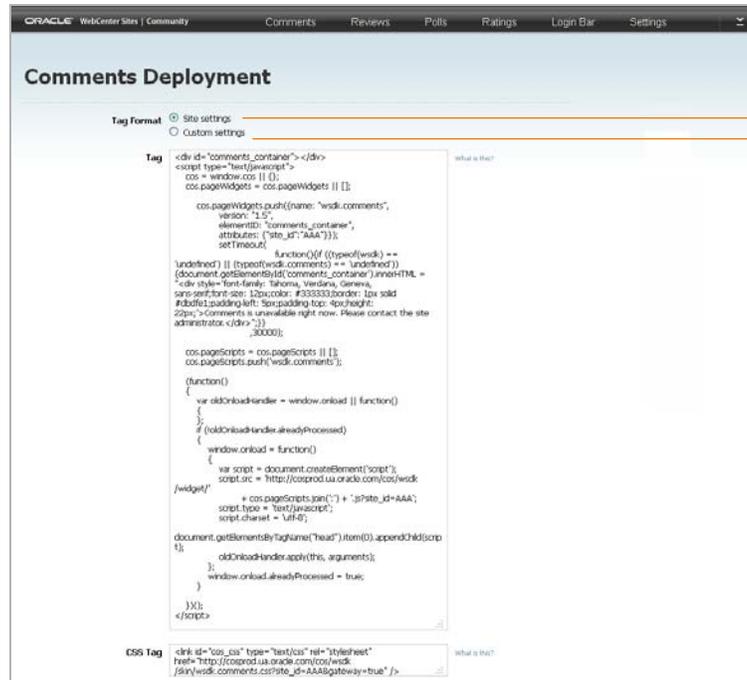
By default, the Recently Commented widget lists topics that have an associated Comments widget of resource type “Others.” To configure the Recently Commented widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type”: “others” parameter (in [line 8](#) of the Recently Commented widget tag). For more information, see “[Recently Commented Widget Tag](#),” on [page 203](#).

- **Most Commented** – to deploy the Most Commented widget tag. When deployed, this widget displays a list of the top ten web pages (topics) to which visitors frequently post comments.

#### Note

By default, the Most Commented widget lists topics that have an associated Comments widget of resource type “Others.” To configure the Most Commented widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type”: “others” parameter (in [line 8](#) of the Most Commented widget tag). For more information, see “[Most Commented Widget Tag](#),” on [page 206](#).

The deployment screen for each widget tag looks similar to the “Comments Deployment” screen, shown below:



**Site settings** is used to deploy the widget tag with its default settings.

**Custom settings** enables you to customize the tag directly in the tag generator. Because the settings are not saved, a deployed custom tag cannot be dynamically updated on the pages where it is displayed. It must be redeployed. For more information, see [“Deploying Custom Comments Widget Tags,”](#) on page 56.

2. In the “Tag Format” field, make sure **Site settings** is selected.
3. Deploy the widget tag:
  - a. In the “Tag” field, copy (**Ctrl+C**) the widget tag.
  - b. Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For detailed instructions, see [steps 4 – 8](#) on [page 29](#) in the section [“How Do I Deploy Community Widget Tags?”](#)

### Hiding the Default Comments Widget

If you want to hide a default Comments widget that is displayed on various web pages, instead of accessing the template’s source code, you (depending on your role in the Community application) can use the Comments widget’s “Permissions” and “Appearance” screens.

1. If you are an administrator or moderator, hide the comment field:
  - a. In the menu bar, select **Comments > Configure > Permissions**.
  - b. In the “Commenting Enabled” field, select **No**.
  - c. Click **Save**.  
The comment field is now hidden from site visitors.
2. If you are an administrator or designer, hide the list of comments:
  - a. In the menu bar, select **Comments > Configure > Appearance**.
  - b. In the “Show Comments” field, select **Do not display**.
  - c. Click **Save**.  
The list of comments is now hidden from site visitors.

## Deploying Custom Comments Widget Tags

You can customize any widget tag related to commenting functionality (except Links to Topics) by settings its properties using the tag generator in the widget's "Deployment" screen. Each time a widget tag is customized, it must be redeployed because custom values cannot be saved in the Community interface; instead they are stored only in that particular instance of the widget tag.

### To configure and deploy custom comments widgets

1. In the menu bar, select **Comments > Deploy > *name of the widget tag you wish to customize and deploy***:
  - **Comments** – to customize and deploy the Comments widget tag; that is,
    - (1) Configure the Comments widget tag with a unique identifier and resource type so that it can be referred to by other commenting related widgets for its content, such as the number of comments and date of the most recently posted comment.
    - (2) Specify the web page (topic title and URL) on which the Comments widget tag is deployed.
    - (3) Reconfigure certain properties that were set as defaults in the tag's "Permissions" screen (**Comments > Configure > Permissions**) and "Appearance" screen (**Comments > Configure > Appearance**).
  - **Comments Summary** – to customize and deploy the Comments Summary widget tag; that is, have the Comments Summary widget tag display the date of the most recently posted comment along with the number of comments posted to a certain topic (web page). The Comments Summary widget tag obtains comment summary information from the Comments widget that is deployed on the desired topic (web page). If the Comments Summary widget tag and the Comments widget tag are on different pages, the Comments Summary widget must be configured to refer to the Comments widget by its resource ID.
  - **Recently Commented** – to customize and deploy the Recently Commented widget tag; that is, have the Recently Commented widget list topics of a desired type (instead of topics of type "Others") to which site visitors have recently posted comments. For example, topics of type "Article" are posted on a web page; each article has its own Comments widget of resource type "Article." If the Recently Commented widget is to list topics of type "Article" to which site visitors have recently posted comments, it must be configured to refer to the Comments widget by resource type.

### Note

To configure the Recently Commented widget to list topics of all resource types, you will have to modify the widget tag by removing the "resource\_type": "others" parameter (in [line 8](#) of the Recently Commented widget tag). For more information, see "[Recently Commented Widget Tag](#)," on page 203.

- **Most Commented** – to customize and deploy the Most Commented widget tag; that is, have the Most Commented widget list the most popular topics of a desired type (instead of topics of type “Others”). For example, topics of type “Article” are posted on a web page; each article has its own Comments widget of resource type “Article.” If the Most Commented widget is to list the most popular topics of type “Article,” it must be configured to refer to the Comments widget by resource type.

#### Note

To configure the Most Commented widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type”: “others” parameter (in [line 8](#) of the Most Commented widget tag). For more information, see “[Most Commented Widget Tag](#),” on [page 206](#).

2. In the “Tag Format” field, select **Custom settings**. Customize the desired widget tag by setting values for the available fields in the tag’s custom deployment screen.

For field definitions, see:

- [Comments > Deploy > Comments > Custom Settings](#)
- [Comments > Deploy > Comments Summary](#)
- [Comments > Deploy > Recently Commented](#)
- [Comments > Deploy > Most Commented](#)

3. Deploy the custom tag:

- a. In the “Tag” field, copy (**Ctrl+C**) the widget tag.
- b. Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For instructions, see [steps 4 – 8 on page 29](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

## Custom Widget Tag Deployment Screens

Once you access a given widget's custom deployment screen, you can use the available fields to set custom (temporary) values for the widget tag. These values are not saved, except in the template to which you copy the customized instance of the tag. This section provides field definitions.

### Comments > Deploy > Comments > Custom Settings

This screen is used to define custom values for a particular instance of the Comments widget tag, which can then be deployed on web pages.

**Comments Deployment**

Tag Format:  Site settings  Custom settings

Comment Type:

Resource ID:

Resource Type:

Comment Submit Box:

Display Order:

Comment Ranking:

Resource Title:

Resource URL:

Post Status:

Login Bar Required:

**Tag**

```
<div id="comments_container_other"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || [];

cos.pageWidgets.push({name: "wsk.comments",
version: "1.5",
elementID: "comments_container_other",
attributes:
{"resource_type":"other","comment_post_status":"open","site_id":"AAA"}
});

setTimeout(
function(){if ((typeof(wsk) ==
undefined) || (typeof(wsk.comments) == undefined))
(document.getElementById("comments_container_other").innerHTML
+ "<div style='font-family:Tahoma,Verdana,Geneva,
sans-serif;font-size:12px;color:#333333;border:1px solid
#d9d9d9;padding-left:5px;padding-top:4px;height:
22px;>Comments is unavailable right now. Please contact the site
administrator.</div>");
},30000);

cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsk.comments");

(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://cosprod.us.oracle.com/cos/wsk/
/widget/
+ cos.pageScripts.join('.') + ".js?site_id=AAA";
script.type = "text/javascript";
script.charset = "utf-8";
document.getElementsByTagName("head").item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
};
window.onload.alreadyProcessed = true;
}
})();
</script>
```

**What is this?**

**CSS Tag**

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsk/
/skin/wsk.comments.css?site_id=AAA&gateway=true" />
```

**What is this?**

**Server-Side Tag for SEO**

```
<%@page import="com.fatwire.cos.widget.tag.CosWidgetTag"%>
<%
CosWidgetTag.builder().widget("wsk.comments").siteName("AAA").
resourceId("1").cssUrl("http://cosprod.us.oracle.com/
/cos").writer(out).request(request).deploy();%>
```

**What is this?**

**Widget Tag** [Download the SEO widget file](#) **What is this?**

**CSS Tag for SEO**

```
<link type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsk/
/skin/wsk.comments.css?site_id=AAA" />
```

**What is this?**

**Note**

The **Default** field value references the setting that is specified in the widget's "Permissions" or "Appearance" screen. If you choose a default setting for your custom tag, and you deploy the tag, the default setting will be dynamically updated if its value is modified and saved in the Community interface.

**Table 10:** Customizing and Deploying the Comments Widget Tag

Field	Description
<b>Comment Type</b>	Select whether your deployment supports a <b>Flat</b> commenting structure, or a <b>Threaded</b> commenting structure which allows replies to comments.
<b>Resource ID</b>	<p>Enter a resource ID (unique identifier) for this instance of the Comments widget tag. This resource ID can be used in other comments widget tags (for example, the Comments Summary widget tag) to refer to this instance of the Comments widget. You can enter a maximum of 50 English alphanumeric characters into this field.</p> <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>• This field is required only if you want to configure other comments widget tags to refer to this Comments widget's content (for example, the number of comments posted to the web page).</li> <li>• If you wish to link the Links to Topics widget to this instance of the Comments widget, you will have to note the URL of the web page on which this Comments widget is deployed.</li> </ul>
<b>Resource Type</b>	<p>Select a resource type (topic type) for this instance of the Comments widget tag. The resource type of a Comments widget tag categorizes the web page on which it is deployed. If you do not specify a resource type, the topic (web page) on which this instance of the Comments widget is deployed is associated with the default "Others" resource type.</p> <p><b>Note:</b> The current category file ("<a href="#">Comments &gt; Configure &gt; Appearance</a>") determines the resource types that are available to you.</p>
<b>Comment Submit Box</b>	Specify whether the comment field will be displayed <b>At the top</b> or <b>At the bottom</b> of the list of comments on a given page.
<b>Display Order</b>	<p>Select one of the following to specify the order in which approved comments will be displayed on a given page:</p> <ul style="list-style-type: none"> <li>• <b>Most ranked at the top</b> – Displays the comments that site visitors frequently rank at the top of the list of comments.</li> <li>• <b>Oldest at the top</b> – Displays the oldest comments at the top of the list of comments.</li> <li>• <b>Newest at the top</b> – Displays the most recently posted comments at the top of the list of comments.</li> <li>• <b>Least ranked at the top</b> – Displays the comments that site visitors ranked the least (or have not ranked) at the top of the list of comments.</li> </ul>

**Table 10:** Customizing and Deploying the Comments Widget Tag *(continued)*

Field	Description
<b>Comment Ranking</b>	Specify whether visitors can rank individual comments as <b>Helpful</b> or <b>Not Helpful</b> . Select one of the following: <ul style="list-style-type: none"> <li>• <b>Enabled for all visitors</b> – All visitors (authenticated and anonymous) can rank the comments posted to a web page.</li> <li>• <b>Enabled for registered visitors</b> – Only authenticated visitors can rank the comments posted to a web page.</li> <li>• <b>Disabled</b> – Individual comments cannot be ranked.</li> </ul>
<b>Resource Title</b>	Enter the title of the web page on which this instance of the Comments widget tag will be deployed.
<b>Resource URL</b>	Enter the URL of the web page on which this instance of the Comments widget tag will be deployed.
<b>Post Status</b>	Select either <b>Open</b> or <b>Close</b> to specify whether visitors can post comments to the page on which this instance of the Comments widget tag is deployed.
<b>Login Bar Required</b>	Select either <b>Yes</b> or <b>No</b> to specify whether a Login Bar widget tag will be embedded in and deployed with this instance of the Comments widget tag.  <b>Note:</b> If you select <b>No</b> , you can still deploy a standalone Login Bar widget tag on the same page as the Comments widget (for instructions, see <a href="#">“Deploying a Login Bar Widget Tag,”</a> on page 140).

## Comments > Deploy > Comments Summary

This screen is used to define custom values for a particular instance of the Comments Summary widget tag, which can then be deployed on web pages.

**Table 11:** Customizing and Deploying the Comments Summary Widget Tag

Field	Description
<b>Resource ID</b>	<p>This field is required only if you are deploying the Comments Summary widget tag on a web page different from the one on which the Comments widget is deployed.</p> <p>Specify the resource ID (unique identifier) of a Comments widget tag. When the Comments Summary widget is deployed, it displays summarized information about the number of comments posted to the web page on which the specified Comments widget is deployed.</p>
<b>Show Last Comment Date</b>	<p>Select <b>Yes</b> to display the most recent date a comment was posted to a given page, or select <b>No</b> to display only the number of comments posted to the page.</p> <p><b>Default Value:</b> No</p>

## Comments > Deploy > Recently Commented

This screen is used to define custom values for a particular instance of the Recently Commented widget tag, which can then be deployed on web pages.

The screenshot shows the 'Recently Commented Deployment' configuration page. The 'Resource Type' is set to 'Others', 'Number of Topics' is 10, and 'Include Since' is empty. The 'Tag' field contains the following JavaScript code:

```

<div id="recently_commented_other_container"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || [];
cos.pageWidgets.push({name: "wsdk.topics",
version: "1.5",
elementID: "_recently_commented_other_container",
attributes:
{"content_type": "recently_commented", "resource_type": "other", "site_id": "AAA", "count_topics": "10"}});
setTimeout(
function(){if (typeof(wsdk) ==
undefined) || (typeof(wsdk.topics) == "undefined"){
(document.getElementById("_recently_commented_other_container")
).innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
sans-serif;font-size: 12px;color: #333333;border: 1px solid
#888; padding-left: 5px;padding-top: 4px; height: 22px;'> Topics is
unavailable right now. Please contact the site administrator.</div>";}
,30000);

cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsdk.topics");

(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://cosprod.us.oracle.com/cos/wsdk
/widget/
+ cos.pageScripts.join("&") + '?site_id=AAA';
script.type = "text/javascript";
script.charset = "utf-8";

document.getElementsByTagName("head").item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
window.onload.alreadyProcessed = true;
};
})();
})();
</script>

```

The 'CSS Tag' field contains the following CSS code:

```

<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsdk
/skin/wsdk.topics.css?site_id=AAA&gateway=true"/>

```

**Table 12:** Customizing and Deploying the Recently Commented Widget Tag

Field	Description
<b>Resource Type</b>	Select a resource type (topic type). When the Recently Commented widget is deployed, it lists the recently commented topics of a certain type (for example “Article”) on which a Comments widget assigned to the specified resource type is deployed. If the deployed Comments widgets are not assigned a resource type, select <b>Others</b> . <b>Default Value:</b> Others <b>Note:</b> The current category file (“ <a href="#">Comments &gt; Configure &gt; Appearance</a> ”) determines the resource types that are available to you.
<b>Number of Topics</b>	Specify the number of web pages (topics) to be included in the list (for example, enter 5 to list the five topics to which site visitors have recently posted comments). <b>Default Value:</b> 10

**Table 12:** Customizing and Deploying the Recently Commented Widget Tag  
(continued)

Field	Description
<b>Include Since</b>	Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Recently Commented widget lists only the web pages whose comments were posted on or after the specified date. This allows you to avoid displaying outdated pages and comments. <b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).

## Comments > Deploy > Most Commented

This screen is used to define custom values for a particular instance of the Most Commented widget tag, which can then be deployed on web pages.

The screenshot shows the 'Most Commented Deployment' configuration page. The 'Resource Type' is set to 'Others', 'Number of Topics' is 10, and 'Include Since' is empty. The 'Tag' field contains the following JavaScript code:

```
<div id="most_commented_other_container"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || {};
cos.pageWidgets.push({name: "wsdk.topics",
version: "1.5",
elementID: "most_commented_other_container",
attributes:
{"content_type": "most_commented", "resource_type": "other", "site_id": "AAA", "count_topics": "10"};
setTimeout(
function(){if (typeof(wsdk) ==
undefined) || (typeof(wsdk.topics) == "undefined"){
(document.getElementById("most_commented_other_container").innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva, sans-serif;font-size: 12px;color: #333333;border: 1px solid #d0d0d0;padding-left: 5px;padding-top: 4px;height: 22px;'> Topics is unavailable right now. Please contact the site administrator </div>");
},30000);
cos.pageScripts = cos.pageScripts || {};
cos.pageScripts.push("wsdk.topics");
(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandlerAlreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://coeprod.us.oracle.com/cos/wsdk
/widget/
+ cos.pageScripts.join("&") + ".js?site_id=AAA";
script.type = "text/javascript";
script.charset = "UTF-8";
document.getElementsByTagName("head").item(0).appendChild(script);
};
oldOnloadHandler.apply(this, arguments);
};
window.onloadAlreadyProcessed = true;
})();
</script>
```

The 'CSS Tag' field contains the following CSS code:

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="http://coeprod.us.oracle.com/cos/wsdk
/widget/wsdk.topics.css?site_id=AAA&gateway=true"/>
```

**Table 13:** Customizing and Deploying the Most Commented Widget Tag

Field	Description
<b>Resource Type</b>	Select a resource type (topic type). When the Most Commented widget tag is deployed, it lists the most popular topics of a certain type (for example “Article”) on which a Comments widget assigned to the specified resource type is deployed. If the deployed Comments widgets are not assigned a resource type, select <b>Others</b> . <b>Default Value:</b> Others <b>Note:</b> The current category file (“ <a href="#">Comments &gt; Configure &gt; Appearance</a> ”) determines the resource types that are available to you.
<b>Number of Topics</b>	Specify the number of web pages (topics) to be included in the list (for example, enter 5 to list the five topics to which site visitors frequently post comments). <b>Default Value:</b> 10

**Table 13:** Customizing and Deploying the Most Commented Widget Tag  
(continued)

Field	Description
<b>Include Since</b>	Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Most Commented widget lists only the web pages whose comments were posted on or after the specified date. This allows you to avoid displaying outdated pages and comments. <b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).



## Chapter 4

# Working with the Reviews Widgets

This chapter provides a quick reference for moderating the reviews posted to your site by visitors, configuring the visitor permissions and appearance settings for the Reviews widget, and deploying the Reviews widget (and all related widgets) to a web page with either dynamic (default) or custom settings.

This chapter contains the following sections:

- [Overview](#)
- [Moderating Reviews](#)
- [Reconfiguring the Reviews Widget's Default Settings](#)
- [Deploying the Reviews Widget Tags](#)

## Overview



The **Reviews** menu is used to work with the Reviews widget and all of its related widgets (Reviews Summary, Links to Topics, Reviews Average Rating, Top Ranked Reviews, Recently Reviewed, and Most Reviewed). The “Reviews” menu provides you with options to moderate the reviews that visitors post to the pages of your website, configure visitor permissions and appearance settings for the Reviews widget, and deploy the Reviews widget and all related widgets (with either default or custom settings) to a web page.

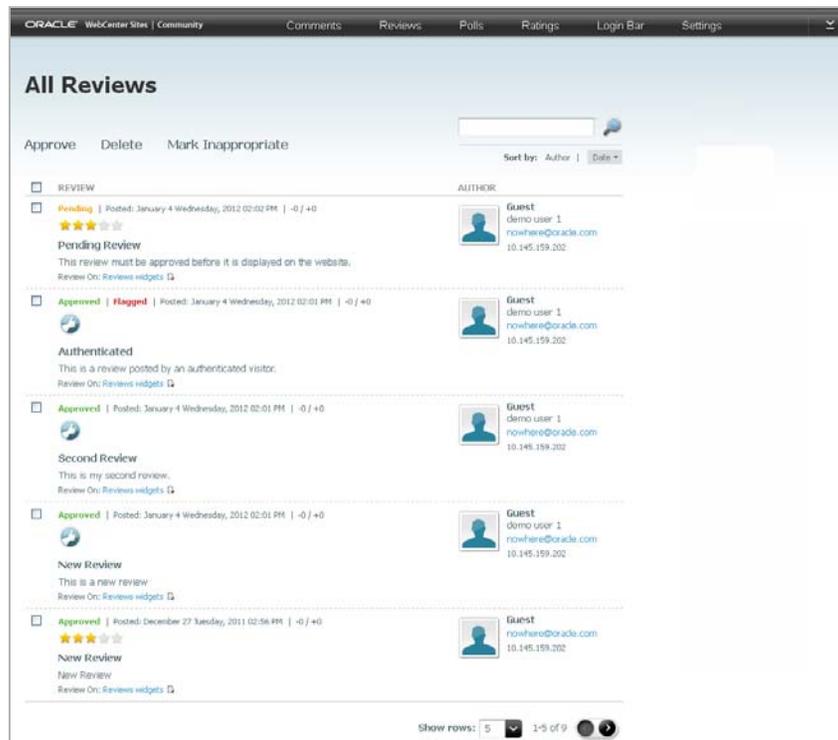
The instructions in this section are written as a quick reference and it is assumed you are already logged in to the Community interface. If you need instructions for logging in, see “[Logging In](#),” on page 26.

## Moderating Reviews

This section is for administrators and moderators. You can access and moderate the reviews that have been posted to your website in the following ways:

### Reviews > Moderate > All Reviews

This screen is used to moderate all of the reviews that have been posted to the pages of your website. The reviews are listed in the order they were posted (most recent to least recent). You can sort the reviews in this list by either the visitor who posted the review or the date the review was posted.



**Table 14:** Moderating All Reviews

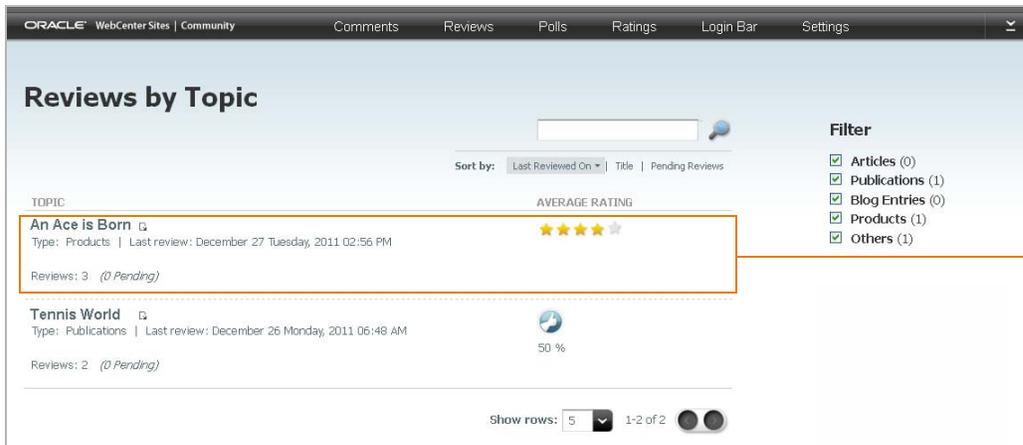
Action	Instructions
<b>Approve visitors' reviews</b>	<ol style="list-style-type: none"> <li>Point to the review you wish to approve.</li> <li>In the menu that appears, select <b>Approve</b>. The approved review is displayed on the website.</li> </ol>
<b>Delete visitors' reviews</b>	<ol style="list-style-type: none"> <li>Point to the review you wish to delete.</li> <li>In the menu that appears, select <b>Delete</b>.</li> <li>In the confirmation box, click <b>OK</b>. The review is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a review as inappropriate</b>	<ol style="list-style-type: none"> <li>Point to the review that contains inappropriate or inaccurate content.</li> <li>In the menu that appears, select <b>Mark Inappropriate</b>. The status of the review changes to <b>Inappropriate</b>. Marking the review as inappropriate indicates to other moderators that the review must be modified before being displayed on the website.</li> </ol> <p><b>Note:</b> Reviews can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, "Reconfiguring General Community Widget Settings."</a></p>
<b>Modify a review</b>	<ol style="list-style-type: none"> <li>Point to the review you wish to modify.</li> <li>In the menu that appears, select <b>Edit</b>. The "Edit Review" screen is displayed.</li> <li>In the "Review" field, make the desired changes to the content.</li> <li>Click <b>Save</b>. The modified review is displayed in the Community interface and on the web page to which it was posted.</li> </ol> <p><b>Note:</b> If manual moderation is enabled ("<a href="#">Settings &gt; Moderation</a>"), manually approve the review you modified to display it on the web page.</p>
<b>Add an authenticated visitor to the "Visitor Whitelist"</b>	<ol style="list-style-type: none"> <li>Point to a review posted by the authenticated visitor you wish to add to the whitelist.</li> <li>In the menu that appears, click <b>Whitelist Author</b>. The "Visitor Whitelist" screen opens, and the visitor's name is displayed in the list of whitelisted visitors.</li> </ol> <p><b>Note:</b> For more information about the "Visitor Whitelist," see "<a href="#">Settings &gt; Visitor Whitelist</a>," on page 157.</p>

**Table 14:** Moderating All Reviews (*continued*)

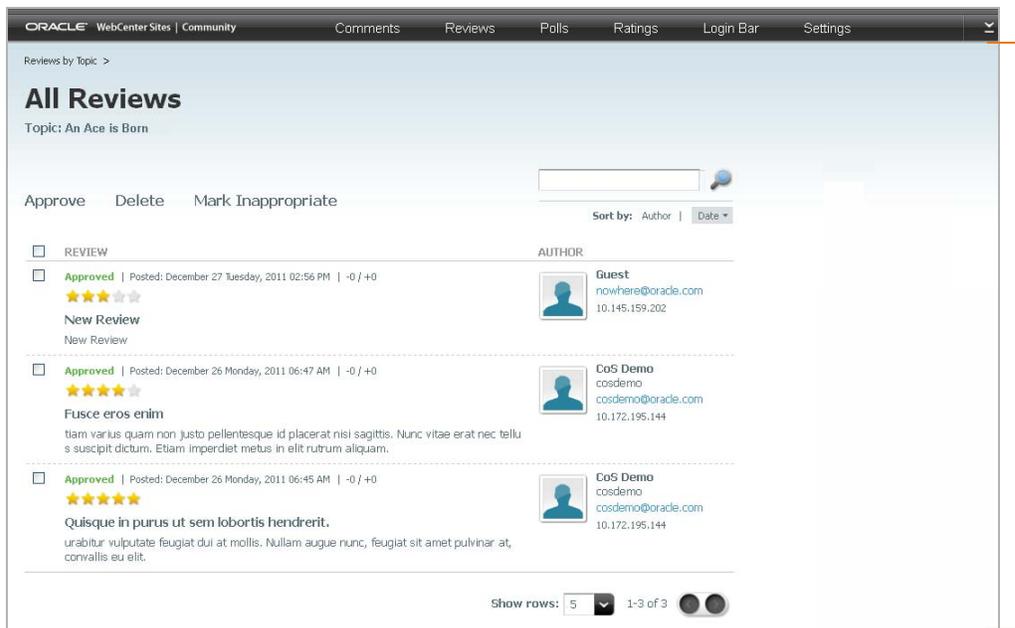
Action	Instructions
<b>Add a visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>1. Point to a review posted by the visitor you wish to add to the blacklist.</li> <li>2. In the menu that appears, click <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens. If the visitor is authenticated, the visitor’s user name is displayed in the list of blacklisted visitors. If the visitor is anonymous, the visitor’s IP address is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Blacklist,” see <a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155.</p>
<b>Search for a review containing specific content</b>	<ol style="list-style-type: none"> <li>1. In the “Search” field, enter the criteria you wish to search for.</li> <li>2. Click the <b>magnifying glass</b> () icon. Only reviews containing the specified search criteria are listed in the “All Reviews” screen.</li> </ol>
<b>Identify a flagged review</b>	A review that is flagged on the website will appear in the “All Reviews” screen with <b>Flagged</b> posted next to its <b>Approved</b> status.
<b>View a topic on the website</b>	<ol style="list-style-type: none"> <li>1. In the list of reviews, navigate to the review that is posted to the desired topic.</li> <li>2. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed Reviews widget, and the reviews visitors posted to the page.</li> </ol>
<b>View all reviews posted by a specific visitor</b>	<ol style="list-style-type: none"> <li>1. In the list of reviews, navigate to the review that is posted by the desired authenticated visitor.</li> <li>2. Click the visitor’s name. The “User Reviews” screen opens and displays only the reviews posted by the selected visitor.</li> </ol> <p><b>Note:</b> For information about moderating reviews by user, see <a href="#">“Reviews &gt; Moderate &gt; Reviews by User,”</a> on page 75.</p>
<b>View all reviews posted to a specific topic</b>	<ol style="list-style-type: none"> <li>1. In the list of reviews, navigate to the review that is posted to the desired topic.</li> <li>2. Click the name of the topic. The “All Reviews” screen opens and displays only the reviews posted to the selected topic.</li> </ol> <p><b>Note:</b> For information about moderating reviews by topic, see <a href="#">“Reviews &gt; Moderate &gt; Reviews by Topic,”</a> on page 71.</p>

## Reviews > Moderate > Reviews by Topic

This screen is used to sort all reviews that have been posted to the pages of your website by the title of the topic (web page) to which the reviews are posted. The average rating given to each topic is listed in the “Average Rating” column, located to the right of the corresponding topic. When you select the title of a topic from this screen, the “All Reviews” screen opens and displays only the reviews posted to that topic.



Clicking the name of a topic in the “Reviews by Topic” screen opens the “All Reviews” screen, which lists only the reviews posted to that topic.



**Table 15:** Moderating Reviews by Topic

Action	Instructions
<b>Filter topics by category</b>	<ol style="list-style-type: none"> <li data-bbox="651 373 1330 516">1. In the “Filter” panel, select the checkbox(es) of the desired resource type(s). The web pages (topics) with Reviews widgets of the selected resource type(s) are the only web pages listed in the “Reviews by Topic” screen.</li> <li data-bbox="651 527 1330 642">2. Select the name of the topic whose reviews you wish to view. The “All Reviews” screen opens and displays only the reviews posted to the topic you selected.</li> </ol> <p data-bbox="651 653 1330 804"><b>Note:</b> The resource types (topic types) available in the “Filter” panel are dynamically modified by the current category file (“Reviews &gt; Configure &gt; Appearance”). “Others” refers to Reviews widgets (and topics) that do not belong to a resource type.</p>
<b>Approve visitors’ reviews</b>	<ol style="list-style-type: none"> <li data-bbox="651 825 1330 877">1. In the “Reviews by Topic” screen, select the topic whose reviews you wish to moderate.</li> <li data-bbox="651 888 1330 940">2. In the “All Reviews” screen for the topic you selected, point to the review you wish to approve.</li> <li data-bbox="651 951 1330 1014">3. In the menu that appears, select <b>Approve</b>. The approved review is displayed on the website.</li> </ol>
<b>Delete visitors’ reviews</b>	<ol style="list-style-type: none"> <li data-bbox="651 1035 1330 1087">1. In the “Reviews by Topic” screen, select the topic whose reviews you wish to moderate.</li> <li data-bbox="651 1098 1330 1150">2. In the “All Reviews” screen for the topic you selected, point to the review you wish to delete.</li> <li data-bbox="651 1161 1330 1192">3. In the menu that appears, select <b>Delete</b>.</li> <li data-bbox="651 1203 1330 1293">4. In the confirmation box, click <b>OK</b>. The review is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a review as inappropriate</b>	<ol style="list-style-type: none"> <li data-bbox="651 1314 1330 1367">1. In the “Reviews by Topic” screen, select the topic whose reviews you wish to moderate.</li> <li data-bbox="651 1377 1330 1451">2. In the “All Reviews” screen for the topic you selected, point to the review that contains inappropriate or inaccurate content.</li> <li data-bbox="651 1461 1330 1608">3. In the menu that appears, select <b>Mark Inappropriate</b>. The status of the review changes to <b>Inappropriate</b>. Marking the review as inappropriate indicates to other moderators that the review must be modified before being displayed on the website.</li> </ol> <p data-bbox="651 1619 1330 1724"><b>Note:</b> Reviews can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, “Reconfiguring General Community Widget Settings.”</a></p>

**Table 15:** Moderating Reviews by Topic *(continued)*

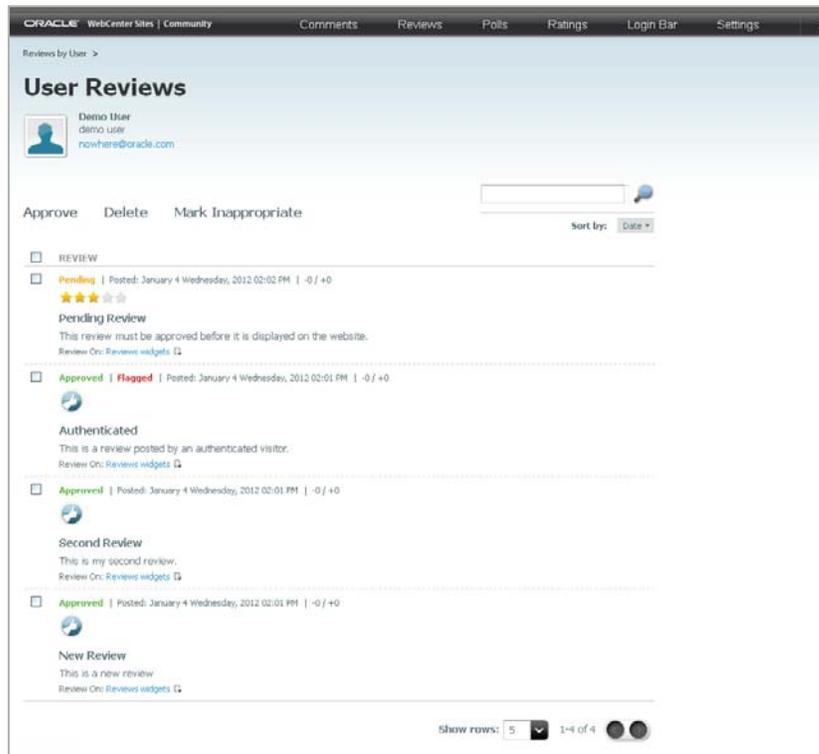
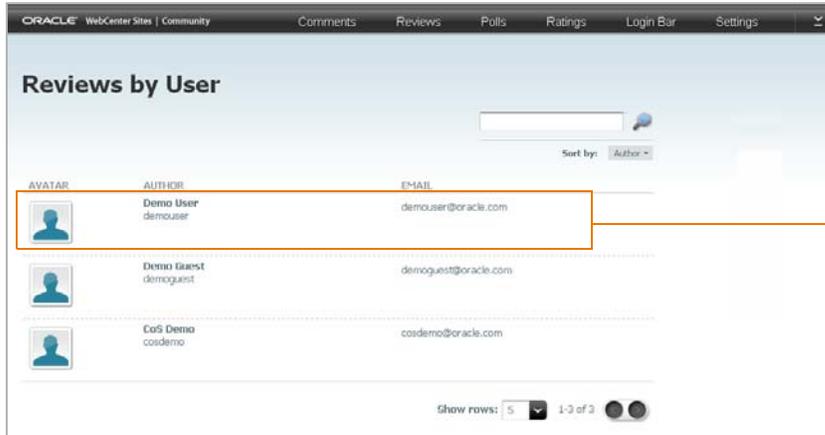
Action	Instructions
<b>Modify a review</b>	<ol style="list-style-type: none"> <li>In the “Reviews by Topic” screen, select the topic whose reviews you wish to moderate.</li> <li>In the “All Reviews” screen for the topic you selected, point to the review you wish to modify.</li> <li>In the menu that appears, select <b>Edit</b>. The “Edit Review” screen is displayed.</li> <li>In the “Review” field, make the desired changes to the content.</li> <li>Click <b>Save</b>. The modified review is displayed in the Community interface and on the web page to which it was posted.</li> </ol> <p><b>Note:</b> If manual moderation is enabled (“<a href="#">Settings &gt; Moderation</a>”), manually approve the review you modified to display it on the web page.</p>
<b>Add an authenticated visitor to the “Visitor Whitelist”</b>	<ol style="list-style-type: none"> <li>In the “Reviews by Topic” screen, select the topic whose reviews you wish to moderate.</li> <li>In the “All Reviews” screen for the topic you selected, point to a review posted by the authenticated visitor you wish to add to the whitelist.</li> <li>In the menu that appears, click <b>Whitelist Author</b>. The “Visitor Whitelist” screen opens, and the visitor’s name is displayed in the list of whitelisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Whitelist,” see “<a href="#">Settings &gt; Visitor Whitelist</a>,” on page 157.</p>
<b>Add a visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>In the “Reviews by Topic” screen, select the topic whose reviews you wish to moderate.</li> <li>In the “All Reviews” screen for the topic you selected, point to a review posted by the visitor you wish to add to the blacklist.</li> <li>In the menu that appears, click <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens. If the visitor is authenticated, the visitor’s user name is displayed in the list of blacklisted visitors. If the visitor is anonymous, the visitor’s IP address is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Blacklist,” see “<a href="#">Settings &gt; Visitor Blacklist</a>,” on page 155.</p>
<b>Search for a specific topic</b>	<ol style="list-style-type: none"> <li>In the “Search” field, enter the title of the topic you wish to search for.</li> <li>Click the <b>magnifying glass</b> () icon. Only topics containing the requested search criteria are listed in the “Reviews by Topic” screen.</li> </ol>

**Table 15:** Moderating Reviews by Topic *(continued)*

Action	Instructions
<b>Identify a flagged review</b>	<ol style="list-style-type: none"><li data-bbox="651 373 1328 426">1. In the “Reviews by Topic” screen, select the topic whose reviews you wish to view.</li><li data-bbox="651 436 1328 552">2. In the list of reviews, identify any flagged reviews. A review that is flagged on the website will appear in the list of reviews with <b>Flagged</b> posted next to its <b>Approved</b> status.</li></ol>
<b>View a topic on the website</b>	<ol style="list-style-type: none"><li data-bbox="651 573 1328 625">1. In the “Reviews by Topic” screen, navigate to the desired topic.</li><li data-bbox="651 636 1328 751">2. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed Reviews widget, and the reviews visitors posted to the page.</li></ol>

## Reviews > Moderate > Reviews by User

This screen is used to sort all posted reviews by the user names of the visitors who posted them. When you select a visitor's name in this screen, the "User Reviews" screen opens and displays the reviews the selected visitor posted along with the visitor's user name, profile picture, and e-mail address (if available).



Clicking a visitor's user name in the "Reviews by User" screen opens the "User Reviews" screen, which displays the selected visitor's profile information along with all the reviews that visitor posted to the website.

**Table 16:** Moderating Reviews by User

Action	Instructions
<b>Approve visitors' reviews</b>	<ol style="list-style-type: none"> <li>1. In the “Reviews by User” screen, select the name of the visitor whose reviews you wish to moderate.</li> <li>2. In the “User Reviews” screen, point to the review you wish to approve.</li> <li>3. In the menu that appears, select <b>Approve</b>. The approved review is displayed on the website.</li> </ol>
<b>Delete visitors' reviews</b>	<ol style="list-style-type: none"> <li>1. In the “Reviews by User” screen, select the name of the visitor whose reviews you wish to moderate.</li> <li>2. In the “User Reviews” screen, point to the review you wish to delete.</li> <li>3. In the menu that appears, select <b>Delete</b>.</li> <li>4. In the confirmation box, click <b>OK</b>. The review is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a review as inappropriate</b>	<ol style="list-style-type: none"> <li>1. In the “Reviews by User” screen, select the name of the visitor whose reviews you wish to moderate.</li> <li>2. In the “User Reviews” screen, point to the review that contains inappropriate or inaccurate content.</li> <li>3. In the menu that appears, select <b>Mark Inappropriate</b>. The status of the review changes to <b>Inappropriate</b>. Marking the review as inappropriate indicates to other moderators that the review must be modified before being displayed on the website.</li> </ol> <p><b>Note:</b> Reviews can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, “Reconfiguring General Community Widget Settings.”</a></p>
<b>Modify a review</b>	<ol style="list-style-type: none"> <li>1. In the “Reviews by User” screen, select the name of the visitor whose reviews you wish to moderate.</li> <li>2. In the “User Reviews” screen, point to the review you wish to modify.</li> <li>3. In the menu that appears, select <b>Edit</b>. The “Edit Review” screen is displayed.</li> <li>4. In the “Review” field, make the desired changes to the content.</li> <li>5. Click <b>Save</b>. The modified review is displayed in the Community interface and on the web page to which it was posted.</li> </ol> <p><b>Note:</b> If manual moderation is enabled (“<a href="#">Settings &gt; Moderation</a>”), manually approve the review to display it on the web page.</p>

**Table 16:** Moderating Reviews by User *(continued)*

Action	Instructions
<b>Add an authenticated visitor to the “Visitor Whitelist”</b>	<ol style="list-style-type: none"> <li>In the “Reviews by User” screen, do one of the following: <ul style="list-style-type: none"> <li>If you know which authenticated visitor you want to add to the whitelist, point to the visitor’s name. Continue to <a href="#">step 2</a>.</li> <li>If you wish to view the reviews posted by the visitor before adding that visitor to the whitelist, do the following: <ol style="list-style-type: none"> <li>Select the name of the authenticated visitor whose reviews you wish to view.</li> <li>In the “User Reviews” screen, point to any of the reviews displayed in the list. Continue to <a href="#">step 2</a>.</li> </ol> </li> </ul> </li> <li>In the menu that appears, select <b>Whitelist Author</b>. The “Visitor Whitelist” screen opens, and the visitor’s name is displayed in the list of whitelisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Whitelist,” see <a href="#">“Settings &gt; Visitor Whitelist,”</a> on page 157.</p>
<b>Add an authenticated visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>In the “Reviews by User” screen, do one of the following: <ul style="list-style-type: none"> <li>If you know which visitor you want to add to the blacklist, point to the visitor’s name. Continue to <a href="#">step 2</a>.</li> <li>If you wish to view the reviews posted by the visitor before adding the visitor to the blacklist, do the following: <ol style="list-style-type: none"> <li>Select the name of the visitor whose reviews you wish to view.</li> <li>In the “User Reviews” screen, point to any of the reviews displayed in the list. Continue to <a href="#">step 2</a>.</li> </ol> </li> </ul> </li> <li>In the menu that appears, select <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens, and the visitor’s name is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Blacklist,” see <a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155.</p>
<b>Search for a specific visitor</b>	<ol style="list-style-type: none"> <li>In the “Search” field, enter the name of the visitor you wish to search for.</li> <li>Click the <b>magnifying glass</b> () icon. Only visitor names that match the specified search criteria are listed in the “Reviews by User” screen.</li> </ol>
<b>Identify a flagged review</b>	<ol style="list-style-type: none"> <li>In the “Reviews by User” screen, select the name of the visitor whose reviews you wish to view.</li> <li>In the “User Reviews” screen, identify any flagged reviews: A review that is flagged on the website will appear in the list of reviews with <b>Flagged</b> posted next to its <b>Approved</b> status.</li> </ol>

**Table 16:** Moderating Reviews by User *(continued)*

Action	Instructions
<b>View a topic on the website</b>	<ol style="list-style-type: none"><li data-bbox="654 369 1295 426">1. In the “Reviews by User” screen, select the name of the visitor whose review is posted to the desired topic.</li><li data-bbox="654 432 1271 489">2. In the “User Reviews” screen, navigate to the desired review.</li><li data-bbox="654 495 1287 621">3. Click the <b>View Topic Page</b>  icon. The web page opens displaying the topic, the deployed Reviews widget, and the reviews visitors posted to the page.</li></ol>

## Reconfiguring the Reviews Widget's Default Settings

Default settings are a widget's saved appearance and permission settings. Administrators and designers can modify the default appearance of the Reviews widget by selecting **Reviews > Configure > Appearance** in the menu bar. Administrators and moderators can reconfigure the default reviewing permissions by selecting **Reviews > Configure > Permissions** in the menu bar. Once the permissions and appearance settings are saved, the system dynamically updates the Reviews widget's default tags in the Community interface and on pages where tags with default settings are deployed.

You can reconfigure the default settings of the Reviews widget in the following ways:

### Reviews > Configure > Appearance

This screen is used to reconfigure the Reviews widget's color schema, sort order, and other display properties.

The screenshot shows the 'Reviews Appearance' configuration page in Oracle WebCenter Sites. The page is divided into several sections:

- GENERAL**: Contains settings for 'Show Reviews' (Display), 'Review Type' (Full review), 'Pagination' (Yes (40 reviews per page)), 'Display Order' (Oldest at the top), 'Review Submit Box' (At the top), 'Review Ranking' (Enabled for registered visitors), 'Skin' (Default), 'Upload Custom CSS' (with a 'Browse...' button and 'Download the current CSS' link), and 'Login Bar Required' (Yes).
- REVIEWING AUTO DISABLE**: Contains 'Max Number of Reviews Allowed' (None) and 'Accept Reviews Until' (None), both with 'What is this?' links.
- REVIEW BODY**: Contains 'Max Length' (450 characters) and 'Type' (Plain text).
- HELP AND DISCLAIMER**: Contains 'Show Help Text' (Display) and 'Show Disclaimer Text' (Display), both with 'What is this?' links. Below each dropdown is a text area containing placeholder text.
- TOPIC CATEGORIES**: Contains 'Upload Categories' (with a 'Browse...' button and 'Download the current category file' link).

At the bottom of the page, there is a 'Save' button with a green checkmark icon.

**Table 17:** Reconfiguring the Reviews Widget's Appearance Settings

Section	Field	Description
General	Show Reviews	Select either <b>Display</b> or <b>Do not Display</b> to specify whether approved reviews are displayed to site visitors.
	Review Type	Select the type of review your deployment supports. Select either: <ul style="list-style-type: none"> <li>• <b>Full review</b> – Supports a review field with a title field and rating capabilities which allows visitors to write a review, include a title for their review, and submit a rating with their review.</li> <li>• <b>Brief review</b> – Supports a review field with rating capabilities which allows visitors to write a review and submit a rating with their review.</li> </ul>
	Pagination	Specify the number of approved reviews that will be displayed on a given page at one time.
	Display Order	Select one of the following to specify the order in which approved reviews will be displayed on a given page: <ul style="list-style-type: none"> <li>• <b>Newest at the top</b> – Displays the most recently posted reviews at the top of the list of reviews.</li> <li>• <b>Oldest at the top</b> – Displays the oldest reviews at the top of the list of reviews.</li> <li>• <b>Least ranked at the top</b> – Displays the reviews that site visitors rank the least (or have not ranked) at the top of the list of reviews.</li> <li>• <b>Most ranked at the top</b> – Displays the reviews that site visitors frequently rank at the top of the list of reviews.</li> </ul>
	Review Submit Box	Specify whether the review field will be displayed <b>At the top</b> or <b>At the bottom</b> of the list of reviews on a given page.
	Review Ranking	Specify whether site visitors can rank individual reviews as <b>Helpful</b> or <b>Not Helpful</b> . Select one of the following: <ul style="list-style-type: none"> <li>• <b>Enabled for all visitors</b> – All visitors (authenticated and anonymous) can rank the reviews posted to a web page.</li> <li>• <b>Enabled for registered visitors</b> – Only authenticated visitors can rank the reviews posted to a web page.</li> <li>• <b>Disabled</b> – Individual reviews cannot be ranked.</li> </ul>
	Skin	Select <b>Default</b> to use the default color schema for the Reviews widget, or select <b>Custom</b> to upload your own CSS file.

**Table 17:** Reconfiguring the Reviews Widget's Appearance Settings *(continued)*

Section	Field	Description
<b>General</b> <i>(continued)</i>	Upload Custom CSS	If you selected the <b>Custom</b> option in the “Skin” field, use the <b>Browse</b> button to select a cascading style sheet (CSS) that will control the appearance of the Reviews widget.
	Login Bar Required	Select either <b>Yes</b> or <b>No</b> to specify whether a Login Bar widget tag will be embedded in and deployed with the Reviews widget tag.  <b>Note:</b> If you select <b>No</b> , you can still deploy a standalone Login Bar widget tag on the same page as the Reviews widget (for instructions, see <a href="#">“Deploying a Login Bar Widget Tag,”</a> on page 140).
<b>Reviewing Auto Disable</b>	Max Number of Reviews Allowed	Enter the maximum number of reviews that can be posted to a given web page. Once the maximum number of reviews is reached, site visitors cannot post reviews to that particular web page.
	Accept Reviews Until	Enter the maximum number of days reviews can be posted to a given web page. Once the specified number of days is reached, site visitors cannot post reviews to that particular web page.
<b>Review Body</b>	Max Length	Select either <b>450 characters</b> or <b>900 characters</b> to set the review field size.
	Type	Select the type of text that can be entered into the review field. Select either: <ul style="list-style-type: none"> <li>• <b>Plain text</b> – Visitors can only enter text into the review field.</li> <li>• <b>HTML with blocked JS</b> – Visitors can enter text and basic HTML tags into the review field.</li> <li>• <b>HTML and Media with blocked JS</b> – Visitors can enter text, basic HTML tags, videos, and so on into the review field.</li> </ul>
<b>Help and Disclaimer</b>	Show Help Text	Select either <b>Display</b> or <b>Do not display</b> to specify whether help text will be displayed to visitors.
	Help Text	If you selected <b>Display</b> in the “Show Help Text” field, enter the help text you wish to display to visitors. When the Reviews widget is deployed, the help text is displayed above the reviews submit box.  <b>Note:</b> This field supports plain text and basic HTML tags. You can use HTML tags to control the appearance of your help text.
	Show Disclaimer Text	Select either <b>Display</b> or <b>Do not display</b> to specify whether a disclaimer will be displayed to visitors.

**Table 17:** Reconfiguring the Reviews Widget's Appearance Settings *(continued)*

Section	Field	Description
<b>Help and Disclaimer</b> <i>(continued)</i>	Disclaimer Text	<p>If you selected <b>Display</b> in the “Show Disclaimer Text” field, enter the disclaimer text you wish to display to visitors. When the Reviews widget is deployed, the disclaimer is displayed beneath the reviews submit box.</p> <p><b>Note:</b> This field supports plain text and basic HTML tags. You can use HTML tags to control the appearance of your disclaimer.</p>
<b>Topic Categories</b>	Upload Categories	<p>Use the <b>Browse</b> button to upload the desired category file, which will dynamically modify the resource types available from:</p> <ul style="list-style-type: none"> <li>• The “Resource Type” field in the Reviews widget tag’s custom deployment screen (“<a href="#">Reviews &gt; Deploy &gt; Reviews &gt; Custom Settings</a>”).</li> <li>• The “Resource Type” field in the Top Ranked Reviews (“<a href="#">Reviews &gt; Deploy &gt; Top Ranked Topics</a>”), Recently Reviewed (“<a href="#">Reviews &gt; Deploy &gt; Recently Reviewed</a>”), and Most Reviewed (“<a href="#">Reviews &gt; Deploy &gt; Most Reviewed</a>”) widget tag deployment screens.</li> <li>• The “Reviews by Topic” screen’s “Filter” panel (“<a href="#">Reviews &gt; Moderate &gt; Reviews by Topic</a>”).</li> </ul> <p><b>Note:</b> The category file must be in .txt format, the resource types listed in the file must be in a key=value structure, each resource type listed in the file must be on its own line, and the size of the file you are uploading can be no more than 20MB.</p>

## Reviews > Configure > Permissions

This screen is used to display or hide the review field and set visitors' permissions to review.

**Table 18:** Reconfiguring Reviewing Permissions for Site Visitors

Section	Field
<b>Reviewing Enabled</b>	<ul style="list-style-type: none"> <li>• <b>Yes</b> configures the widget to display a review field.</li> <li>• <b>No</b> hides the review field, but displays only approved reviews, if the widget was previously enabled.</li> </ul>
<b>Who Can Review?</b>	<p>Available only if the widget is configured to display a review field:</p> <ul style="list-style-type: none"> <li>• <b>Anonymous users can review</b> grants reviewing permissions to all site visitors. <ul style="list-style-type: none"> <li>Available only if anonymous visitors can review: <ul style="list-style-type: none"> <li>- <b>User must enter a name</b> requires anonymous visitors to specify a name when posting a review.</li> <li>- <b>User must enter an email address</b> requires anonymous visitors to specify an email address when posting a review.</li> <li>- <b>User must enter a captcha</b> requires anonymous visitors to enter the letters and/or numbers of a captcha image before they can post a review.</li> </ul> <p><b>Note:</b> By default, this option is greyed out (not available). For information about enabling the “User must enter a captcha” option, see the <i>Oracle WebCenter Sites Developer’s Guide for the Community Application</i>.</p> <ul style="list-style-type: none"> <li>- <b>Restrict user to one review per topic</b> allows visitors to post only one review to a given web page (topic).</li> </ul> </li> </ul> </li> <li>• <b>Only authenticated users can review</b> grants reviewing permissions only to authenticated visitors. <ul style="list-style-type: none"> <li>- <b>User can modify and delete his reviews</b> allows authenticated visitors to modify and delete the reviews they post.</li> <li>- <b>Restrict user to one review per topic</b> allows authenticated visitors to post only one review to a given web page (topic).</li> </ul> </li> </ul>

## Deploying the Reviews Widget Tags

This section is for administrators and designers. You can deploy the Reviews widget tag and all of its related widget tags with either default or custom settings by selecting **Reviews > Deploy > *name of the widget tag you wish to deploy*** in the menu bar. [Table 19](#) lists the reviews widget tags and indicates which type of deployment each tag supports.

**Table 19:** Deploying the Reviews Widget Tags

Widget Tag	Deploy with Default Settings	Configure and Deploy Custom Settings
Reviews	✓	✓
Reviews Summary	✓ <sup>a</sup>	✓
Links to Topics	✓ <sup>b</sup>	
Average Rating		✓ <sup>c</sup>
Top Ranked Reviews	✓	✓
Recently Reviewed	✓	✓
Most Reviewed	✓	✓

- a. If you deploy this widget tag (Reviews Summary) on a web page different from the one on which the desired Reviews widget is deployed, you will have to link the Reviews Summary widget to the Reviews widget by specifying a resource ID. For more information, see “[Reviews > Deploy > Reviews Summary](#),” on page 95.
- b. This widget tag (Links to Topics) must be linked to the web page that displays the desired topic and its reviews. For more information, see “[Links to Topics Widget Tag](#),” on page 215.
- c. You can configure this widget to display the average rating of multiple web pages. For instructions, see “[Reviews Average Rating Widget Tag](#),” on page 218.

This section contains the following topics:

- [Deploying a Default Reviews Widget Tag](#)
- [Deploying Custom Reviews Widget Tags](#)
- [Custom Widget Tag Deployment Screens](#)

## Deploying a Default Reviews Widget Tag

You can deploy any widget tag related to reviewing functionality with default settings (except the Reviews Average Rating widget tag). The default settings for the Reviews widget are its saved settings, which are configured in the “Permissions” screen (**Reviews > Configure > Permissions**) and “Appearance” screen (**Reviews > Configure > Appearance**). The default settings for all other review related widgets are configured out-of-the-box and cannot be reconfigured in the Community interface. Each reviews widget has its own deployment screen, which displays the widget tag. The default settings are used as property values in the widget tag.

### To deploy default reviews widget tags

1. In the menu bar, select **Reviews > Deploy > *name of the widget tag you wish to deploy***:
  - **Reviews** – to deploy the Reviews widget tag. If reviewing is enabled (**Reviews > Configure > Permissions**), the Reviews widget displays a review field. If displaying reviews is enabled (**Reviews > Configure > Appearance**), then a list of site visitors’ reviews is displayed on the web page.

#### Note

If any of the following deployment scenarios are true, you will have to customize the Reviews widget by specifying a resource ID:

- You wish to link the Links to Topics widget to a Reviews widget.
- You want to deploy the Reviews Summary widget on a web page different from the one on which the desired Reviews widget is deployed.
- You want the Reviews Average Rating widget to display the average rating of the web page on which a Reviews widget tag is deployed.

For more information, see the description for the [Resource ID](#) field in [Table 20, on page 93](#).

- **Reviews Summary** – to deploy the Reviews Summary widget tag. When deployed on a web page displaying a Reviews widget, the Reviews Summary widget displays short, summarized information about the number of reviews posted to the web page (topic).

#### Note

If you deploy this widget (Reviews Summary) on a web page different from the one on which the desired Reviews widget is deployed, you will have to link the Reviews Summary widget to the Reviews widget by specifying a resource ID. For more information, see the description for the [Resource ID](#) field in [Table 21, on page 95](#).

- **Links to Topics** – to deploy the Links to Topics widget tag. When deployed, the Links to Topics widget displays the number of reviews posted to a given topic (web page) along with a link to that page.

#### Note

The Links to Topics widget tag must be linked to the web page that displays the desired topic and its reviews. The link must include the resource ID of the desired topic's associated Reviews widget. For more information, see [line 1](#) in the section “[Links to Topics Widget Tag](#),” on [page 215](#).

- **Top Ranked Topics** – to deploy the Top Ranked Reviews widget tag. When deployed, this widget lists and provides links to the website's highest rated web pages (topics).

#### Note

By default, the Top Ranked Reviews widget lists topics that have an associated Reviews widget of resource type “Others.” To configure the Top Ranked Reviews widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Top Ranked Reviews widget tag). For more information, see “[Top Ranked Reviews Widget Tag](#),” on [page 222](#).

- **Recently Reviewed** – to deploy the Recently Reviewed widget tag. When deployed, this widget displays a list of the top ten web pages (topics) to which site visitors have recently posted reviews.

#### Note

By default, the Recently Reviewed widget lists topics that have an associated Reviews widget of resource type “Others.” To configure the Recently Reviewed widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Recently Reviewed widget tag). For more information, see “[Recently Reviewed Widget Tag](#),” on [page 225](#).

- **Most Reviewed** – to deploy the Most Reviewed widget tag. When deployed, this widget displays a list of the top ten web pages (topics) to which site visitors frequently post reviews.

### Note

By default, the Most Reviewed widget lists topics that have an associated Reviews widget of resource type “Others.” To configure the Most Reviewed widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type”: “other” parameter (in [line 8](#) of the Most Reviewed widget tag). For more information, see “[Most Reviewed Widget Tag](#),” on [page 228](#).

The deployment screen for each widget looks similar to the “Reviews Deployment” screen, shown below:

**Site settings** is used to deploy the widget tag with its default settings.

**Custom settings** enables you to customize the tag directly in the tag generator. Because the settings are not saved, a deployed custom tag cannot be dynamically updated on the pages where it is displayed. It must be redeployed. For more information, see “[Deploying a Default Reviews Widget Tag](#),” on [page 85](#).

2. In the “Tag Format” field, make sure **Site settings** is selected.
3. Deploy the widget tag:
  - a. In the “Tag” field, copy (**Ctrl+C**) the widget tag.
  - b. Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For detailed instructions, see [steps 4 – 8](#) on [page 29](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

### Hiding the Default Reviews Widget

If you want to hide a default Reviews widget that is displayed on various web pages, instead of accessing the template's source code, you (depending on your role in the Community application) can use the Reviews widget's "Permissions" and "Appearance" screens.

1. If you are an administrator or moderator, hide the review field:
  - a. In the menu bar, select **Reviews > Configure > Permissions**.
  - b. In the "Reviewing Enabled" field, select **No**.
  - c. Click **Save**.  
The review field is now hidden from site visitors.
2. If you are an administrator or designer, hide the list of reviews:
  - a. In the menu bar, select **Reviews > Configure > Appearance**.
  - b. In the "Show Reviews" field, select **Do not display**.
  - c. Click **Save**.  
The list of reviews is now hidden from site visitors.

## Deploying Custom Reviews Widget Tags

You can customize any widget tag related to reviewing functionality (except Links to Topics) by setting its properties using the tag generator in the widget's "Deployment" screen. Each time a widget tag is customized, it must be redeployed because custom values cannot be saved in the Community interface, instead they are saved only in that particular instance of the widget tag.

### To configure and deploy custom reviews widget tags

1. In the menu bar, select **Reviews > Deploy > name of the widget tag you wish to customize and deploy**:
  - **Reviews** – to customize and deploy the Reviews widget tag: that is, (1) Configure the Reviews widget tag with a unique identifier and resource type so that it can be referred to by other reviewing related widgets for its content, such as the number of reviews and the average rating. (2) Specify the web page (topic title and URL) on which the Reviews widget tag is deployed. (3) Reconfigure certain properties that were set as defaults in the tag's "Permissions" (**Reviews > Configure > Permissions**) screen and "Appearance" (**Reviews > Configure > Appearance**) screen.
  - **Reviews Summary** – to customize and deploy the Reviews Summary widget tag: that is, have the Reviews Summary widget tag display the date of the most recent review along with the number of reviews posted to a certain topic (web page). The Reviews Summary widget tag obtains summary information from the Reviews widget deployed on the desired topic (web page). If the Reviews Summary widget tag and the Reviews widget tag are on different pages, the Reviews Summary widget must be configured to refer to the Reviews widget by its resource ID.
  - **Average Rating** – to customize and deploy the Reviews Average Rating widget tag: that is, for the Reviews Average Rating widget to display the average rating of a given topic (web page), the Reviews Average Rating widget must be configured to refer to the Reviews widget, deployed on the desired topic (web page), by its resource ID.

#### Note

You can configure this widget to display the average rating of multiple web pages. For instructions, see "[Reviews Average Rating Widget Tag](#)," on page 218.

- **Top Ranked Topics** – to customize and deploy the Top Ranked Reviews widget tag: that is, have the Top Ranked Reviews widget display the highest rated topics of a desired type (instead of topics of type “Others”). For example, topics of type “Article” are posted on a web page; each article has its own Reviews widget of resource type “Article.” If the Top Ranked Reviews widget is to list the highest rated topics of type “Article,” it must be configured to refer to the Reviews widget by resource type.

#### Note

To configure the Top Ranked Reviews widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Top Ranked Reviews widget tag). For more information, see “[Top Ranked Reviews Widget Tag](#),” on page 222.

- **Recently Reviewed** – to customize and deploy the Recently Reviewed widget tag: that is, have the Recently Reviewed widget display recently reviewed topics of a desired type (instead of topics of type “Others”). For example, topics of type “Article” are posted on a web page; each article has its own Reviews widget of resource type “Article.” If the Recently Reviewed widget is to list the recently reviewed topics of type “Article,” it must be configured to refer to the Reviews widget by resource type.

#### Note

To configure the Recently Reviewed widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Recently Reviewed widget tag). For more information, see “[Recently Reviewed Widget Tag](#),” on page 225.

- **Most Reviewed** – to customize and deploy the Most Reviewed widget tag: that is, have the Most Reviewed widget display the most popular topics of a desired type (instead of topics of type “Others”). For example, topics of type “Article” are posted on a web page; each article has its own Reviews widget of resource type “Article.” If the Most Reviewed widget is to list the most popular topics of type “Article,” it must be configured to refer to the Reviews widget by resource type.

#### Note

To configure the Recently Reviewed widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Most Reviewed widget tag). For more information, see “[Recently Reviewed Widget Tag](#),” on page 225.

2. In the “Tag Format” field, select **Custom Settings**. Customize the desired widget tag by setting values for the available fields in the tag’s custom deployment screen.

For field definitions, see:

- [Reviews > Deploy > Reviews > Custom Settings](#)
  - [Reviews > Deploy > Reviews Summary](#)
  - [Reviews > Deploy > Average Rating](#)
  - [Reviews > Deploy > Top Ranked Topics](#)
  - [Reviews > Deploy > Recently Reviewed](#)
  - [Reviews > Deploy > Most Reviewed](#)
3. Deploy the custom tag:
    - a. In the “Tag” field, copy (**Ctrl+C**) the widget tag.
    - b. Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For instructions, see [steps 4 – 8 on page 29](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

## Custom Widget Tag Deployment Screens

Once you access a given widget's custom deployment screen, you can use the available fields to set custom (temporary) values for the widget tag. These values are not saved, except in the template to which you copy the customized instance of the tag. This section provides field definitions.

### Reviews > Deploy > Reviews > Custom Settings

This screen is used to define custom values for a particular instance of the Reviews widget tag, which can then be deployed on web pages.

**Reviews Deployment**

Tag Format:  Site settings,  Custom settings

Review Type: Default

Rating: Stars

Resource ID:

Resource Type: Others

Review Submit Box: Default

Display Order: Default

Review Ranking: Default

Resource Title:

Resource URL:

Post Status: Open

Login Bar Required: Default

**Tag**

```
<div id="reviews_container_other"></div>
<script type="text/javascript">
  cos = window.cos || {};
  cos.pageWidgets = cos.pageWidgets || [];

  cos.pageWidgets.push({name: "wsdk.reviews",
    version: "1.5",
    elementID: "reviews_container_other",
    attributes:
    { "rating_type": "stars", "resource_type": "other", "review_post_status":
    "open", "site_id": "AAA" });
    setTimeout(
      function() { if ((typeof(wsdk) ==
    'undefined') || (typeof(wsdk.reviews) == 'undefined'))
    (document.getElementById("reviews_container_other").innerHTML
    += "<div style='font-family: Tahoma, Verdana, Geneva,
    sans-serif;font-size: 12px;color: #888888;border: 1px solid
    #dbdfe1;padding-left: 5px;padding-top: 4px;height: 22px;>Reviews
    is unavailable right now. Please contact the site administrator.
    </div> ->"),
    ,30000);

    cos.pageScripts = cos.pageScripts || [];
    cos.pageScripts.push("wsdk.reviews");

    (function()
    {
      var oldOnloadHandler = window.onload || function()
      {
      };
      if (oldOnloadHandler.alreadyProcessed)
      {
        window.onload = function()
        {
          var script = document.createElement("script");
          script.src = "http://cosprod.us.oracle.com/cos/wsdk
          /widget/
          + cos.pageScripts.join(';') + '?site_id=AAA';
          script.type = "text/javascript";
          script.charset = "utf-8";

          document.getElementsByTagName("head").item(0).appendChild(script);
          oldOnloadHandler.apply(this, arguments);
        };
        window.onload.alreadyProcessed = true;
      }
    })();
  </script>
```

**CSS Tag**

```
<link id="cos_css" type="text/css" rel="stylesheet"
  href="http://cosprod.us.oracle.com/cos/wsdk
  /skin/wsdk.reviews.css?site_id=AAA&gateway=true" />
```

**Server-Side Tag for SEO**

```
<%@page import="com.fstwire.cos.widget.tag.CosWidgetTag"%>
<%
  CosWidgetTag.builder().widget("wsdk.reviews").siteName("AAA").res
  ourceId("").cosUrl("http://cosprod.us.oracle.com
  /cos").writer(out).request(request).deploy(); %>
```

**Widget Tag** [Download the SEO widget file](#)

**CSS Tag for SEO**

```
<link type="text/css" rel="stylesheet"
  href="http://cosprod.us.oracle.com/cos/wsdk
  /skin/wsdk.reviews.css?site_id=AAA" />
```

**Note**

The **Default** field value references the setting that is specified in the widget's "Permissions" or "Appearance" screen. If you choose a default setting for your custom tag, and you deploy the tag, that default setting will be dynamically updated if its value is modified and saved in the Community interface.

**Table 20:** Customizing and Deploying the Reviews Widget Tag

Field	Description
<b>Review Type</b>	Select the type of review structure your deployment supports. Select either: <ul style="list-style-type: none"> <li>• <b>Brief review</b> – Supports a review field with rating capabilities, which allows visitors to write a review and submit a rating with their review.</li> <li>• <b>Full review</b> – Supports a review field with a title field and rating capabilities, which allows visitors to write a review, include a title for their review, and submit a rating with their review.</li> </ul>
<b>Rating</b>	Determine whether the Reviews widget supports a binary rating ( <b>Thumbs Up/Thumbs Down</b> ) where visitors can specify whether they liked or disliked the topic, or a complex rating ( <b>Stars</b> ) where visitors can rate the topic by selecting a number of stars.
<b>Resource ID</b>	Enter a resource ID (unique identifier) for this instance of the Reviews widget tag. This resource ID can be used in other reviews widget tags (for example, the Reviews Summary widget tag) to refer to this instance of the Reviews widget. You can enter a maximum of 50 English alphanumeric characters into this field. <p><b>Note:</b></p> <ul style="list-style-type: none"> <li>• This field is required only if you want to configure other reviews widget tags to refer to this Reviews widget's content (for example, the number of reviews posted to the web page).</li> <li>• If you wish to link the Links to Topics widget to this instance of the Reviews widget, you will have to note the URL of the web page on which this Reviews widget is deployed.</li> </ul>
<b>Resource Type</b>	Select a resource type (topic type) for this instance of the Reviews widget tag. The resource type of a Reviews widget tag categorizes the web page on which it is deployed. If you do not specify a resource type, the topic (web page) on which this instance of the Reviews widget tag is deployed is associated with the default "Others" resource type. <p><b>Note:</b> The current category file ("<a href="#">Reviews &gt; Configure &gt; Appearance</a>") determines the resource types that are available to you.</p>
<b>Review Submit Box</b>	Specify whether the review field will be displayed <b>At the top</b> or <b>At the bottom</b> of the list of reviews on a given page.

**Table 20:** Customizing and Deploying the Reviews Widget Tag *(continued)*

Field	Description
<b>Display Order</b>	<p>Select one of the following to specify the order in which approved reviews will be displayed on a given page:</p> <ul style="list-style-type: none"> <li>• <b>Most ranked at the top</b> – Displays the reviews that site visitors rank the most at the top of the list of reviews.</li> <li>• <b>Oldest at the top</b> – Displays the oldest reviews at the top of the list of reviews.</li> <li>• <b>Newest at the top</b> – Displays the most recently posted reviews at the top of the list of reviews.</li> <li>• <b>Least ranked at the top</b> – Displays the reviews site visitors rank the least (or have not ranked) at the top of the list of reviews.</li> </ul>
<b>Review Ranking</b>	<p>Specify whether visitors can rank individual reviews as <b>Helpful</b> or <b>Not Helpful</b>. Select one of the following:</p> <ul style="list-style-type: none"> <li>• <b>Enabled for all visitors</b> – All visitors (authenticated and anonymous) can rank the reviews posted to a web page.</li> <li>• <b>Enabled for registered visitors</b> – Only authenticated visitors can rank the reviews posted to a web page.</li> <li>• <b>Disabled</b> – Individual reviews cannot be ranked.</li> </ul>
<b>Resource Title</b>	Enter the title of the web page (topic) on which the widget tag will be deployed.
<b>Resource URL</b>	Enter the URL of the web page on which the widget tag will be deployed.
<b>Post Status</b>	Select either <b>Open</b> or <b>Close</b> to specify whether site visitors can post reviews to the page on which this widget tag is deployed.
<b>Login Bar Required</b>	<p>Select either <b>Yes</b> or <b>No</b> to specify whether a Login Bar widget tag will be embedded in and deployed with the Reviews widget tag.</p> <p><b>Note:</b> If you select <b>No</b>, you can still deploy a standalone Login Bar widget tag on the same page as the Reviews widget (for instructions, see <a href="#">“Deploying a Login Bar Widget Tag,”</a> on page 140).</p>

## Reviews > Deploy > Reviews Summary

This screen is used to define custom values for a particular instance of the Reviews Summary widget tag, which can then be deployed on web pages.

**Resource ID**

**Show Last Review Date**

**Tag**

```
<div id="reviews_summary_container"></div>
<script type="text/javascript">
  cos = window.cos || {};
  cos.pageWidgets = cos.pageWidgets || {};

  cos.pageWidgets.push({name: "reviews-summary",
    version: "1.1",
    elementID: "reviews_summary_container",
    attributes:
    { "show_last_comment_date": "false", "site_id": "AAA" });
    setTimeout(
      function(){if (typeof(cos) == "undefined"){
        [typeof(cos.reviews-summary) == "undefined"]}
        {document.getElementById("reviews_summary_container").innerHTML =
        "<div style='font-family: Tahoma, Verdana, Geneva, sans-serif;font-size: 12px;color: #808080;border: 1px solid #d0d0d0;padding-left: 5px;padding-top: 4px;height: 20px;'>Reviews Summary is unavailable right now. Please contact the site administrator.</div>?");
        },30000);

    cos.pageScripts = cos.pageScripts || [];
    cos.pageScripts.push("reviews-summary");

    (function()
    {
      var oldOnloadHandler = window.onload || function()
      {
        };
      if (oldOnloadHandler.alreadyProcessed)
      {
        window.onload = function()
        {
          var script = document.createElement("script");
          script.src = "http://cosprod.us.oracle.com/cos/wsdk/
          /widget?
          + cos.pageScripts.join("&") + "&site_id=AAA";
          script.type = "text/javascript";
          script.charset = "utf-8";

          document.getElementsByTagName("head").item(0).appendChild(script);
          oldOnloadHandler.apply(this, arguments);
        };
        window.onload.alreadyProcessed = true;
      }
    }());
  </script>
```

**CSS Tag**

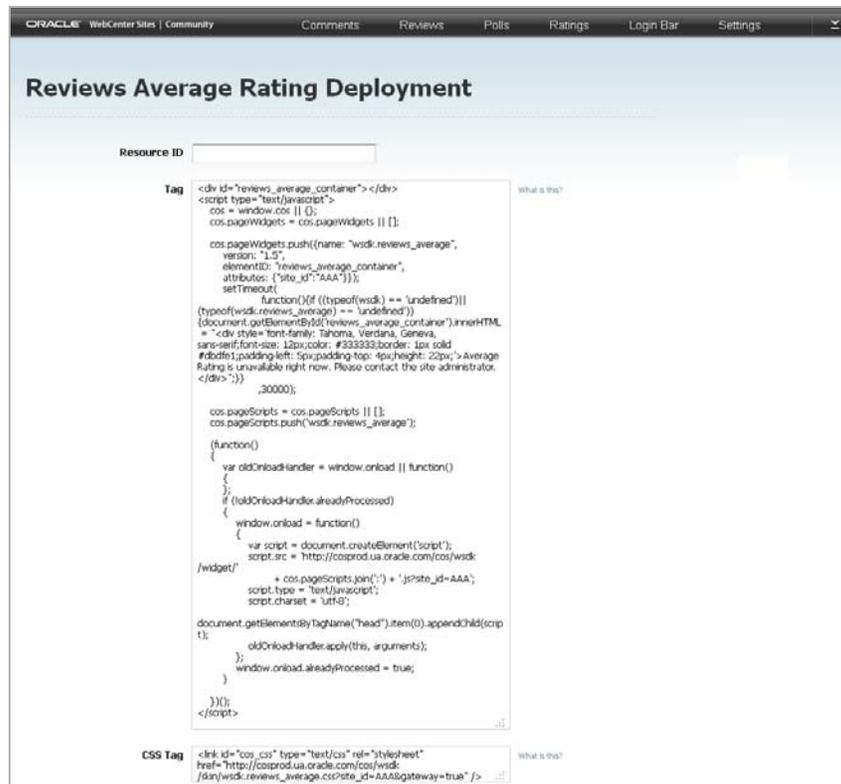
```
<link id="cos_css" type="text/css" rel="stylesheet"
  href="http://cosprod.us.oracle.com/cos/wsdk/skin/reviews-summary.css?site_id=AAA&gateway=true" />
```

**Table 21:** Customizing and Deploying the Reviews Summary Widget Tag

Field	Description
<b>Resource ID</b>	This field is required only if you are deploying the Reviews Summary widget tag on a web page different from the one on which the desired Reviews widget is deployed.  Specify the resource ID (unique identifier) of a Reviews widget tag. When the Reviews Summary widget is deployed, it displays summarized information about the number of reviews posted to the web page on which the specified Reviews widget is deployed.
<b>Show Last Review Date</b>	Select <b>Yes</b> to display the most recent date a review was posted to a web page, or select <b>No</b> to display only the number of reviews posted to the page.  <b>Default Value: No</b>

## Reviews > Deploy > Average Rating

This screen is used to define custom values for a particular instance of the Reviews Average Rating widget tag, which can then be deployed on web pages.



**Table 22:** Customizing and Deploying the Average Rating Widget Tag

Field	Description
<b>Resource ID</b>	<p>Specify the resource ID (unique identifier) of a Reviews widget tag. When this instance of the Reviews Average Rating widget is deployed, it displays the average rating of the web page on which the specified Reviews widget is deployed.</p> <p><b>Note:</b> To configure the Reviews Average Rating widget to display the average rating of multiple pages, see “<a href="#">Reviews Average Rating Widget Tag</a>,” on page 218.</p>

## Reviews > Deploy > Top Ranked Topics

This screen is used to define custom values for a particular instance of the Top Ranked Reviews widget tag, which can then be deployed on web pages.

The screenshot shows the 'Top Ranked Reviews Deployment' configuration page. The form includes the following fields:

- Resource Type:** A dropdown menu set to 'Others'.
- Number of Topics:** A text input field containing the value '10'.
- Include Since:** An empty text input field.

Below the form is a large text area containing the JavaScript and CSS code for the widget tag. The JavaScript code includes logic for pushing widget data and handling on-load events. The CSS code defines the appearance of the widget.

**Table 23:** Customizing and Deploying the Top Ranked Reviews Widget Tag

Field	Description
<b>Resource Type</b>	Select a resource type (topic type). When the Top Ranked Reviews widget is deployed, it lists the highest rated topics of a certain type (for example, “Article”) on which a Reviews widget assigned to the specified resource type is deployed. If the deployed Reviews widgets are not assigned a resource type, select <b>Others</b> . <b>Default Value:</b> Others <b>Note:</b> The current category file (“ <a href="#">Reviews &gt; Configure &gt; Appearance</a> ”) determines the resource types that are available to you.
<b>Number of Topics</b>	Specify the number of web pages (topics) to be included in the list (for example, enter 5 to display a list of the five highest rated topics). <b>Default Value:</b> 10

**Table 23:** Customizing and Deploying the Top Ranked Reviews Widget Tag  
(continued)

Field	Description
<b>Include Since</b>	<p>Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Top Ranked Reviews widget lists only the web pages whose reviews were posted on or after the specified date. This allows you to avoid displaying outdated pages and reviews.</p> <p><b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).</p>

## Reviews > Deploy > Recently Reviewed

This screen is used to define custom values for a particular instance of the Recently Reviewed widget tag, which can then be deployed on web pages.

The screenshot shows the 'Recently Reviewed Deployment' configuration page. The 'Resource Type' is set to 'Others', 'Number of Topics' is 10, and 'Include Since' is empty. The 'Tag' field contains the following JavaScript code:

```

<div id="_recently_reviewed_other_container"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || [];
cos.pageWidgets.push({name: "wsdk:topics",
  version: "1.5",
  elementID: "_recently_reviewed_other_container",
  attributes:
  {content_type: "recently_reviewed", resource_type: "other", site_id: "AAA", count_topics: "10"}});
setTimeout(
function(){if (typeof(wsdk) ==
undefined) || (typeof(wsdk.topics) == "undefined"){
(document.getElementById("_recently_reviewed_other_container").innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva, sans-serif;font-size: 12px;color: #333333;border: 1px solid #000;padding-left: 5px;padding-top: 45px;height: 220px;'> Topics is unavailable right now. Please contact the site administrator.</div ');}
,30000);

cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsdk:topics");

(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandlerAlreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://cooprod.us.oracle.com/cos/wsdk/widgetU"
+ cos.pageScripts.join(';') + ".js?site_id=AAA";
script.type = "text/javascript";
script.charset = "utf8";

document.getElementsByTagName("head").item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
};
window.onloadAlreadyProcessed = true;
}
})();
</script>

```

The 'CSS Tag' field contains the following CSS link:

```

<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cooprod.us.oracle.com/skin/wsdk/skin/wsdk:topics.css?site_id=AAA&gateway=true"/>

```

**Table 24:** Customizing and Deploying the Recently Reviewed Widget Tag

Field	Description
<b>Resource Type</b>	<p>Select a resource type (topic type). When the Recently Reviewed widget is deployed, it lists the recently reviewed topics of a certain type (for example, "Article") on which a Reviews widget assigned to the specified resource type is deployed. If the deployed Reviews widgets are not assigned a resource type, select <b>Others</b>.</p> <p><b>Default Value:</b> Others</p> <p><b>Note:</b> The current category file ("<a href="#">Reviews &gt; Configure &gt; Appearance</a>") determines the resource types that are available to you.</p>
<b>Number of Topics</b>	<p>Specify the number of web pages (topics) to be included in the list (for example, enter 5 to display the five most recently reviewed topics).</p> <p><b>Default Value:</b> 10</p>

**Table 24:** Customizing and Deploying the Recently Reviewed Widget Tag  
(continued)

Field	Description
<b>Include Since</b>	<p>Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Recently Reviewed widget lists only the web pages whose reviews were posted on or after the specified date. This allows you to avoid displaying outdated pages and reviews.</p> <p><b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).</p>

## Reviews > Deploy > Most Reviewed

This screen is used to define custom values for a particular instance of the Most Reviewed widget tag, which can then be deployed on web pages.

The screenshot shows the 'Most Reviewed Deployment' configuration page. The 'Resource Type' dropdown is set to 'Others'. The 'Number of Topics' input field contains the value '10'. The 'Include Since' field is empty. The 'Tag' field contains a large block of JavaScript code that initializes the widget, sets a timeout, and loads a CSS file. The 'CSS Tag' field contains a CSS link tag for the widget's styles.

**Table 25:** Customizing and Deploying the Most Reviewed Widget Tag

Field	Description
Resource Type	Select a resource type (topic type). When the Most Reviewed widget is deployed, it lists the most reviewed topics of a certain type (for example, “Article”) on which a Reviews widget assigned to the specified resource type is deployed. If the deployed Reviews widgets are not assigned a resource type, select <b>Others</b> . <b>Default Value:</b> Others <b>Note:</b> The current category file (“ <a href="#">Reviews &gt; Configure &gt; Appearance</a> ”) determines the resource types that are available to you.
Number of Topics	Specify the number of web pages (topics) to be included in this list (for example, enter 5 to list the five topics site visitors frequently review). <b>Default Value:</b> 10

**Table 25:** Customizing and Deploying the Most Reviewed Widget Tag *(continued)*

Field	Description
<b>Include Since</b>	Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Most Reviewed widget lists only the web pages whose reviews were posted on or after the specified date. This enables you to avoid displaying outdated pages and reviews. <b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).

## Chapter 5

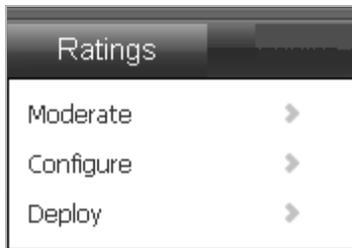
# Working with Ratings Widgets

This chapter provides a quick reference for moderating the ratings posted to your site by visitors, configuring visitors' rating permissions and the appearance of the ratings widgets, and deploying the ratings widgets with either default or custom settings to a web page.

This chapter contains the following sections:

- [Overview](#)
- [Moderating Ratings](#)
- [Reconfiguring the Ratings Widget's Default Settings](#)
- [Deploying the Ratings Widgets](#)

## Overview



The **Ratings** menu is used to work with all widgets related to rating functionality (Stars Ratings, Thumbs Up/Down Ratings, Like It Ratings, Recommend Ratings, Ratings Average Rating, Recently Rated, and Most Rated). The “Ratings” menu provides you with options to moderate the ratings that visitors post to web pages, configure visitor permissions and appearance settings for the ratings widgets, and deploy ratings widgets (with either default or custom settings) to web pages.

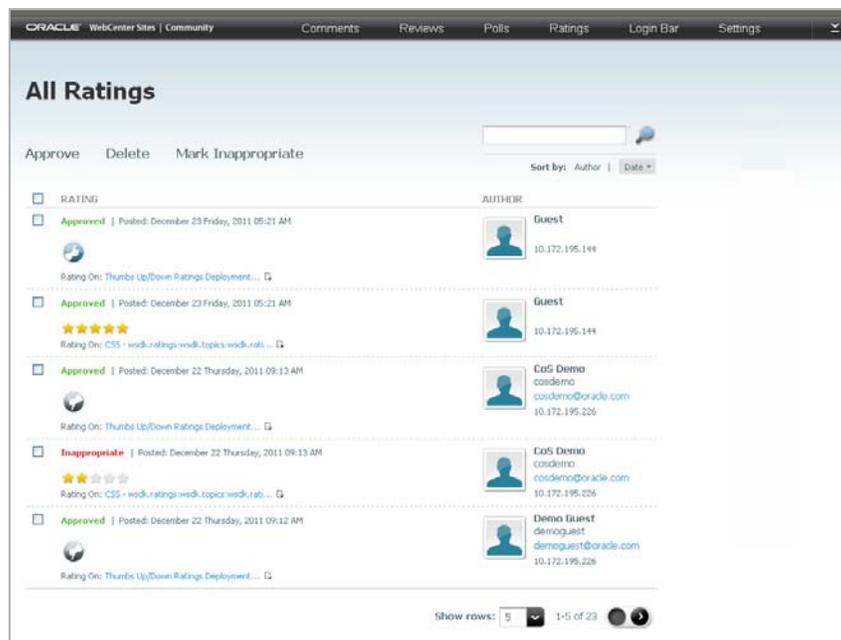
The instructions in this section are written as a quick reference and it is assumed you are already logged in to the Community interface. If you need instructions for logging in, see “[Logging In](#),” on page 26.

## Moderating Ratings

This section is for administrators and moderators. You can access and moderate the ratings that site visitors give to the content of your website in the following ways:

### Ratings > Moderate > All Rating

This screen is used to view and moderate all of the ratings that visitors have given to your website’s content. The type of rating that site visitors can give to the content of your website depends on the type of ratings widget that was deployed: Stars, Thumbs Up/Down, Like It, and Recommend Ratings. The ratings are listed in the order they were posted (most recent to least recent). You can sort the ratings in this list by either the visitor who posted the rating or the date the rating was posted.



**Table 26:** Moderating All Ratings

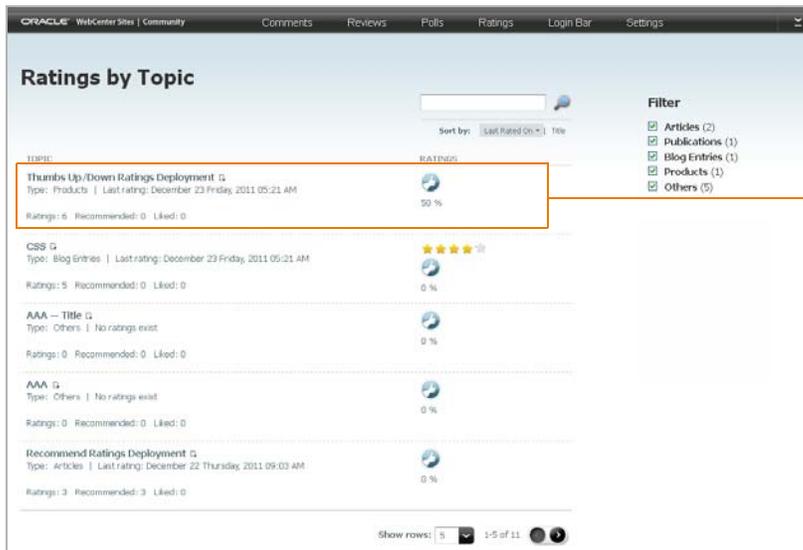
Action	Instructions
<b>Approve visitors' ratings</b>	<ol style="list-style-type: none"> <li>1. Point to the rating you wish to approve.</li> <li>2. In the menu that appears, select <b>Approve</b>. The approved rating is displayed on the website.</li> </ol>
<b>Delete visitors' ratings</b>	<ol style="list-style-type: none"> <li>1. Point to the rating you wish to delete.</li> <li>2. In the menu that appears, select <b>Delete</b>.</li> <li>3. In the confirmation box, click <b>OK</b>. The rating is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a rating as inappropriate</b>	<ol style="list-style-type: none"> <li>1. Point to the rating that is inappropriate or inaccurate.</li> <li>2. In the menu that appears, select <b>Mark Inappropriate</b>. The status of the rating changes to <b>Inappropriate</b>. Marking the rating as inappropriate indicates to other moderators that the rating must be reviewed before displaying it on the website.</li> </ol> <p><b>Note:</b> Ratings can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, "Reconfiguring General Community Widget Settings."</a></p>
<b>Add an authenticated visitor to the "Visitor Whitelist"</b>	<ol style="list-style-type: none"> <li>1. Point to the rating posted by the visitor you wish to add to the whitelist.</li> <li>2. In the menu that appears, select <b>Whitelist Author</b>. The "Visitor Whitelist" screen opens, and the visitor's name is displayed in the list of whitelisted visitors.</li> </ol> <p><b>Note:</b> For more information about the "Visitor Whitelist," see <a href="#">"Settings &gt; Visitor Whitelist,"</a> on page 157.</p>
<b>Add a visitor to the "Visitor Blacklist"</b>	<ol style="list-style-type: none"> <li>1. Point to the rating posted by the visitor you wish to add to the blacklist.</li> <li>2. In the menu that appears, select <b>Blacklist Author</b>. The "Visitor Blacklist" screen opens. If the visitor is authenticated, the visitor's user name is displayed in the list of blacklisted visitors. If the visitor is anonymous, the visitor's IP address is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the "Visitor Blacklist," see <a href="#">"Settings &gt; Visitor Blacklist,"</a> on page 155.</p>
<b>Search for ratings posted by a specific visitor</b>	<ol style="list-style-type: none"> <li>1. In the "Search" field, enter the name or IP address of the visitor you wish to search for.</li> <li>2. Click the <b>magnifying glass</b> () icon. All ratings posted by visitors matching your search criteria are listed in the "All Ratings" screen.</li> </ol>
<b>View a topic on the website</b>	<ol style="list-style-type: none"> <li>1. In the list of ratings, navigate to the rating posted to the desired topic.</li> <li>2. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed ratings widget, and the ratings visitors posted to the page.</li> </ol>

**Table 26:** Moderating All Ratings

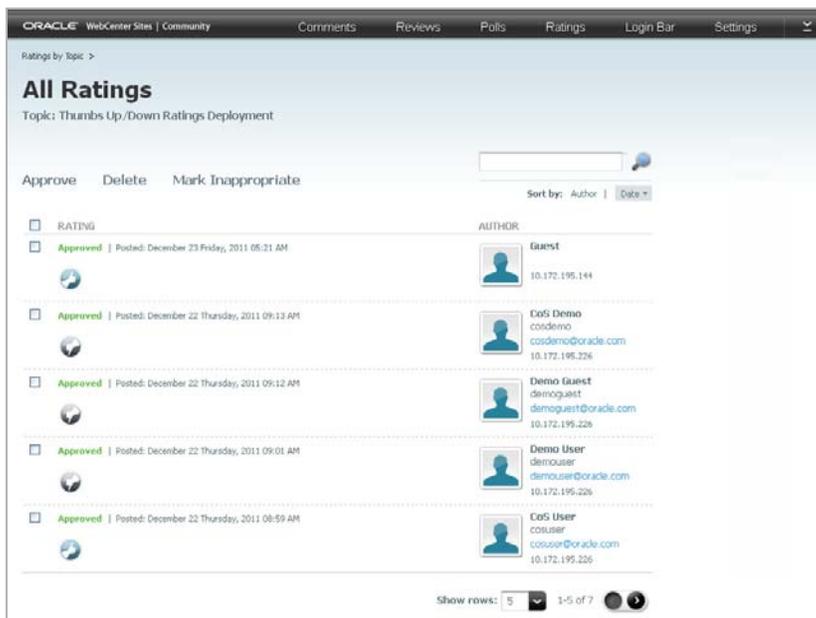
Action	Instructions
<p><b>View all ratings posted by a specific visitor</b></p>	<ol style="list-style-type: none"> <li data-bbox="651 373 1331 426">1. In the list of ratings, navigate to the rating that is posted by the desired authenticated visitor.</li> <li data-bbox="651 436 1331 531">2. Click the visitor's name. The "User Ratings" screen opens and displays only the ratings posted by the selected visitor.</li> </ol> <p data-bbox="651 531 1331 594"><b>Note:</b> For information about moderating ratings by user, see "<a href="#">Ratings &gt; Moderate &gt; Ratings by User</a>," on page 110.</p>
<p><b>View all ratings posted to a specific topic</b></p>	<ol style="list-style-type: none"> <li data-bbox="651 615 1331 667">1. In the list of ratings, navigate to the rating that is posted to the desired topic.</li> <li data-bbox="651 678 1331 772">2. Click the name of the topic. The "All Ratings" screen opens and displays only the ratings posted to the selected topic.</li> </ol> <p data-bbox="651 772 1331 835"><b>Note:</b> For information about moderating ratings by topic, see "<a href="#">Ratings &gt; Moderate &gt; Ratings by Topic</a>," on page 107.</p>

## Ratings > Moderate > Ratings by Topic

This screen is used to sort all ratings given to the pages of your website by the title of the topic (page) to which the rating is posted. The average rating given to each topic is listed in the “Ratings” column, located to the right of the corresponding topic. When you select the title of a topic from this screen, the “All Ratings” screen opens and displays only the ratings posted to that topic.



Clicking the name of a topic in the “Ratings by Topic” screen opens the “All Ratings” screen, which displays only the ratings posted to that topic.



**Table 27: Moderating Ratings by Topic**

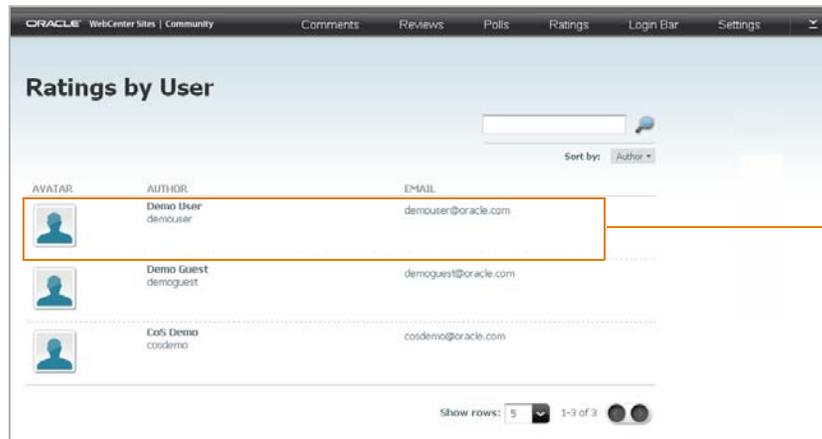
Action	Instructions
<b>Filter topics by category</b>	<ol style="list-style-type: none"> <li data-bbox="651 373 1326 516">1. In the “Filter” panel, select the checkbox(es) of the desired resource type(s). The web pages (topics) with ratings widgets of the selected resource type(s) are the only web pages that will be displayed in the “Ratings by Topic” screen.</li> <li data-bbox="651 527 1326 617">2. Select the topic whose ratings you wish to view. The “All Ratings” screen opens and displays only the ratings posted to the topic you selected.</li> </ol> <p data-bbox="651 627 1326 747"><b>Note:</b> The resource types available in the “Filter” panel are dynamically modified by the current category file (“<a href="#">Ratings &gt; Configure &gt; Appearance</a>”). “Others” refers to ratings widgets (and topics) that do not belong to a resource type.</p>
<b>Approve visitors’ ratings</b>	<ol style="list-style-type: none"> <li data-bbox="651 768 1326 821">1. In the “Ratings by Topic” screen, select the topic whose ratings you wish to moderate.</li> <li data-bbox="651 831 1326 884">2. In the “All Ratings” screen for the topic you selected, point to the rating you wish to approve.</li> <li data-bbox="651 894 1326 957">3. In the menu that appears, select <b>Approve</b>. The approved rating is displayed on the website.</li> </ol>
<b>Delete visitors’ ratings</b>	<ol style="list-style-type: none"> <li data-bbox="651 978 1326 1031">1. In the “Ratings by Topic” screen, select the topic whose ratings you wish to moderate.</li> <li data-bbox="651 1041 1326 1094">2. In the “All Ratings” screen for the topic you selected, point to the rating you wish to delete.</li> <li data-bbox="651 1104 1326 1136">3. In the menu that appears, select <b>Delete</b>.</li> <li data-bbox="651 1146 1326 1230">4. In the confirmation box, click <b>OK</b>. The rating is removed from the Community interface and from the web page to which it was posted.</li> </ol>
<b>Mark a rating as inappropriate</b>	<ol style="list-style-type: none"> <li data-bbox="651 1251 1326 1304">1. In the “Ratings by Topic” screen, select the topic whose ratings you wish to moderate.</li> <li data-bbox="651 1314 1326 1367">2. In the “All Ratings” screen for the topic you selected, point to the rating that is inappropriate or inaccurate.</li> <li data-bbox="651 1377 1326 1520">3. In the menu that appears, select <b>Mark Inappropriate</b>. The status of the rating changes to <b>Inappropriate</b>. Marking the rating as inappropriate indicates to other moderators that the rating must be reviewed before displaying it on the website.</li> </ol> <p data-bbox="651 1530 1326 1635"><b>Note:</b> Ratings can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, “Reconfiguring General Community Widget Settings.”</a></p>

**Table 27:** Moderating Ratings by Topic *(continued)*

Action	Instructions
<b>Add an authenticated visitor to the “Visitor Whitelist”</b>	<ol style="list-style-type: none"> <li data-bbox="651 373 1326 430">1. In the “Ratings by Topic” screen, select the topic whose ratings you wish to moderate.</li> <li data-bbox="651 436 1326 514">2. In the “All Ratings” screen for the topic you selected, point to the rating posted by the visitor you wish to add to the whitelist.</li> <li data-bbox="651 520 1326 619">3. In the menu that appears, select <b>Whitelist Author</b>. The “Visitor Whitelist” screen opens, and the visitor’s name is displayed in the list of whitelisted visitors.</li> </ol> <p data-bbox="651 625 1326 682"><b>Note:</b> For more information about the “Visitor Whitelist,” see <a href="#">“Settings &gt; Visitor Whitelist,”</a> on page 157.</p>
<b>Add a visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li data-bbox="651 705 1326 762">1. In the “Ratings by Topic” screen, select the topic whose ratings you wish to moderate.</li> <li data-bbox="651 768 1326 846">2. In the “All Ratings” screen for the topic you selected, point to the rating posted by the visitor you wish to add to the blacklist.</li> <li data-bbox="651 852 1326 1035">3. In the menu that appears, select <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens. If the visitor is authenticated, the visitor’s user name is displayed in the list of blacklisted visitors. If the visitor is anonymous, the visitor’s IP address is displayed in the list of blacklisted visitors.</li> </ol> <p data-bbox="651 1041 1326 1098"><b>Note:</b> For more information about the “Visitor Blacklist,” see <a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155.</p>
<b>Search for a specific topic</b>	<ol style="list-style-type: none"> <li data-bbox="651 1115 1326 1171">1. In the “Search” field, enter the title of the topic you wish to search for.</li> <li data-bbox="651 1178 1326 1272">2. Click the <b>magnifying glass</b> () icon. The topic you searched for is the only topic listed in the “Ratings by Topic” screen.</li> </ol>
<b>View a topic on the website</b>	<ol style="list-style-type: none"> <li data-bbox="651 1289 1326 1346">1. In the “Ratings by Topic” screen, navigate to the desired topic.</li> <li data-bbox="651 1352 1326 1446">2. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed ratings widget, and the ratings visitors posted to the page.</li> </ol>

## Ratings > Moderate > Ratings by User

This screen is used to sort all ratings posted to your website by the names of the visitors who posted them. When you select a visitor's name from this screen, the "User Ratings" screen opens and displays the ratings that visitor posted to the website along with the visitor's user name, profile picture, and e-mail address (if available).



Clicking a visitor's user name in the "Ratings by User" screen opens the "User Ratings" screen, which displays the selected visitor's profile information along with all the ratings that visitor has posted to the website.



**Table 28:** Moderating Ratings by User

Action	Instructions
Approve visitors' ratings	<ol style="list-style-type: none"> <li>1. In the "Ratings by User" screen, select the name of the visitor whose ratings you wish to moderate.</li> <li>2. In the "User Ratings" screen, point to the rating you wish to approve.</li> <li>3. In the menu that appears, select <b>Approve</b>. The approved rating is displayed on the website.</li> </ol>

**Table 28:** Moderating Ratings by User *(continued)*

Action	Instructions
<b>Delete visitors' ratings</b>	<ol style="list-style-type: none"> <li>1. In the "Ratings by User" screen, select the name of the visitor whose ratings you wish to moderate.</li> <li>2. In the "User Ratings" screen, point to the rating you wish to delete.</li> <li>3. In the menu that appears, select <b>Delete</b>.</li> <li>4. In the confirmation box, click <b>OK</b>.</li> </ol> <p>The rating is removed from the Community interface and from the web page to which it was posted.</p>
<b>Mark a rating as inappropriate</b>	<ol style="list-style-type: none"> <li>1. In the "Ratings by User" screen, select the name of the visitor whose ratings you wish to moderate.</li> <li>2. In the "User Ratings" screen, point to the rating that is inappropriate or inaccurate.</li> <li>3. In the menu that appears, click <b>Mark Inappropriate</b>.</li> </ol> <p>The status of the rating changes to <b>Inappropriate</b>. Marking the rating as inappropriate indicates to other moderators that the rating must be reviewed before displaying it on the website.</p> <p><b>Note:</b> Ratings can be marked as inappropriate either manually or through auto moderation. For information about configuring automatic moderation, see <a href="#">Chapter 8, "Reconfiguring General Community Widget Settings."</a></p>
<b>Add an authenticated visitor to the "Visitor Whitelist"</b>	<ol style="list-style-type: none"> <li>1. In the "Ratings by User" screen, do one of the following: <ul style="list-style-type: none"> <li>- If you know which authenticated visitor you want to add to the whitelist, point to the visitor's name. Continue to <a href="#">step 2</a>.</li> <li>- If you wish to view the ratings posted by the visitor before adding that visitor to the whitelist, do the following: <ol style="list-style-type: none"> <li>a. Select the name of the authenticated visitor whose ratings you wish to view.</li> <li>b. In the "User Ratings" screen, point to any of the ratings displayed in the list. Continue to <a href="#">step 2</a>.</li> </ol> </li> </ul> </li> <li>2. In the menu that appears, select <b>Whitelist Author</b>.</li> </ol> <p>The "Visitor Whitelist" screen opens, and the visitor's name is displayed in the list of whitelisted visitors.</p> <p><b>Note:</b> For more information about the "Visitor Whitelist," see <a href="#">"Settings &gt; Visitor Whitelist,"</a> on page 157.</p>

**Table 28:** Moderating Ratings by User *(continued)*

Action	Instructions
<b>Add an authenticated visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>1. In the “Ratings By User” screen, do one of the following: <ul style="list-style-type: none"> <li>- If you know which visitor you want to add to the blacklist, point to the visitor’s name. (Continue to <a href="#">step 2.</a>)</li> <li>- If you wish to view the ratings posted by the visitor before adding that visitor to the blacklist, do the following: <ol style="list-style-type: none"> <li>a. Select the name of the visitor whose ratings you wish to view.</li> <li>b. In the “User Ratings” screen, point to any of the ratings displayed in the list. (Continue to <a href="#">step 2.</a>)</li> </ol> </li> </ul> </li> <li>2. In the menu that appears, select <b>Blacklist Author</b>. The “Visitor Blacklist” screen opens, and the visitor’s name is displayed in the list of blacklisted visitors.</li> </ol> <p><b>Note:</b> For more information about the “Visitor Blacklist,” see <a href="#">“Settings &gt; Visitor Blacklist,”</a> on page 155.</p>
<b>Search for a specific user</b>	<ol style="list-style-type: none"> <li>1. In the “Search” field, enter the name of the visitor you wish to search for.</li> <li>2. Click the <b>magnifying glass</b> () icon. The names of the visitors that match your search criteria are the only names displayed in the “Ratings by User” screen.</li> </ol>
<b>View a topic on the website</b>	<ol style="list-style-type: none"> <li>1. In the “Ratings by User” screen, select the name of the visitor whose rating is posted to the desired topic.</li> <li>2. In the “User Ratings” screen, navigate to the desired rating.</li> <li>3. Click the <b>View Topic Page</b> () icon. The web page opens displaying the topic, the deployed ratings widget, and the ratings visitors posted to the page.</li> </ol>

## Reconfiguring the Ratings Widget's Default Settings

Default settings are a widget's saved appearance and permission settings. Administrators and designers can modify the default appearance settings of the ratings widgets (Stars, Thumbs Up/Down, Like It, and Recommend) by selecting **Ratings > Configure > Appearance** in the menu bar. Administrators and moderators can reconfigure the default rating permissions by selecting **Ratings > Configure > Permissions** in the menu bar. Once the permissions and settings are saved, the system dynamically updates the default ratings widget tags in the Community interface and on pages where tags with default settings are deployed.

You can reconfigure the default settings of the ratings widgets (Stars, Thumbs Up/Down, Like It, and Recommend) in the following ways:

### Ratings > Configure > Appearance

This screen can be used by administrators and designers to reconfigure the Stars, Thumbs Up/Down, Like It, and Recommend Ratings widgets' display properties, such as whether a Login Bar widget will be displayed with each type of ratings widget tag you deploy.

**Table 29:** Reconfiguring the Ratings Widget's Appearance Settings

Section	Field	Description
General	Login Bar Required	Select either <b>Yes</b> or <b>No</b> to specify whether a Login Bar widget tag will be embedded in and deployed with each type of ratings widget tag (Stars, Thumbs Up/Down, Like It, and Recommend). <b>Note:</b> If you select <b>No</b> , you can still deploy a standalone Login Bar widget tag on the same page as a ratings widget (for instructions, see <a href="#">“Deploying a Login Bar Widget Tag,”</a> on page 140).

**Table 29:** Reconfiguring the Ratings Widget's Appearance Settings *(continued)*

Section	Field	Description
<b>Topic Categories</b>	Upload Categories	<p>Use the <b>Browse</b> button to upload the desired category file, which will dynamically modify the resource types available from:</p> <ul style="list-style-type: none"> <li>• The “Resource Type” field in the Stars, Thumbs Up/Down, Like It, and Recommend Ratings widget tag custom deployment screens.</li> <li>• The “Resource Type” field in the Recently Rated (“<a href="#">Ratings &gt; Deploy &gt; Recently Rated</a>”) and Most Rated (“<a href="#">Ratings &gt; Deploy &gt; Most Rated</a>”) widget tag deployment screens.</li> <li>• The “Ratings by Topics” screen’s “Filter” panel (“<a href="#">Ratings &gt; Moderate &gt; Ratings by Topic</a>”).</li> </ul> <p><b>Note:</b> The category file must be in <code>.txt</code> format, the resource types listed in the file must be in a <code>key=value</code> structure, each resource type listed in the file must be on its own line, and the size of the file you are uploading can be no more than 20MB.</p>

## Ratings > Configure > Permissions

This screen can be used by administrators and moderators to set visitors' permissions to rate website content.

**Table 30:** Reconfiguring Rating Permissions for Site Visitors

Section	Field
Who Can Rate	<ul style="list-style-type: none"> <li>• <b>Anonymous users can rate</b> grants rating permissions to all site visitors.</li> <li>• <b>Only authenticated users can rate</b> grants rating permissions only to authenticated site visitors.</li> </ul>

## Deploying the Ratings Widgets

This section is for administrators and designers. You can deploy the widget tags related to rating functionality with either default or custom settings by selecting **Ratings > Deploy > name of the widget tag you wish to deploy** in the menu bar. [Table 31](#) lists the ratings widget tags and indicates which type of deployment each tag supports.

**Table 31:** Deploying the Ratings Widget Tags

Widget Tag	Deploy with Default Settings	Configure and Deploy Custom Settings
Stars Ratings	✓	✓
Thumbs Up/Down Ratings	✓	✓
Like It Ratings	✓	✓
Recommend Ratings	✓	✓
Average Rating		✓ <sup>a</sup>
Recently Rated	✓	✓
Most Rated	✓	✓

a. You can configure this widget to display the average rating of multiple web pages. For instructions, see [“Ratings Average Rating Widget Tag,” on page 243](#).

This section contains the following topics:

- [Deploying a Default Ratings Widget Tag](#)
- [Deploying Custom Ratings Widget Tags](#)
- [Custom Ratings Widget Tag Deployment Screens](#)

## Deploying a Default Ratings Widget Tag

You can deploy any widget tag related to rating functionality with default settings (except the Ratings Average Rating widget tag). The default settings for the Stars, Thumbs Up/Down, Like It, and Recommend ratings widgets are their saved settings, which are configured in the **Ratings** menu's "Permissions" screen (**Ratings > Configure > Permissions**) and "Appearance" screen (**Ratings > Configure > Appearance**). The default settings for all other ratings widgets are configured out-of-the-box and cannot be reconfigured in the Community interface.

Each ratings widget has its own deployment screen, which displays the widget tag. The default settings are used as property values in the widget tag.

### To deploy default ratings widget tags

1. In the menu bar, select **Ratings > Deploy > name of the widget tag you wish to deploy**:
  - **Stars Ratings** – to deploy the Stars Ratings widget tag. When this tag is deployed, it displays a rating field of five stars which site visitors can use to rate the content of a web page (one star represents the lowest rating that can be given to a web page and five stars represents the highest rating).

#### Note

If you want the Ratings Average Rating widget to display the average rating of the web page on which a Stars Ratings widget is deployed, you will have to customize the Reviews widget by specifying a resource ID. For more information, see the description of the [Resource ID](#) field in [Table 32, on page 124](#).

- **Thumb Up/Down Ratings** – to deploy the Thumbs Up/Down Ratings widget tag. When this tag is deployed, it displays a rating field with a thumbs up and a thumbs down option. By clicking one of these options, visitors can indicate whether a particular topic was helpful.
- **Like It Ratings** – to deploy the Like It Ratings widget tag. When this tag is deployed, it displays a **Like** option which visitors can click to indicate whether they "liked" a particular topic.
- **Recommend Ratings** – to deploy the Recommend Ratings widget tag. When this tag is deployed, it displays an **Email a Friend** option which visitors can click to email the URL of the page on which this widget is deployed to one or more of their contacts.

- **Recently Rated** – to deploy the Recently Rated widget tag. When this tag is deployed, it displays a list of the top ten web pages (topics) site visitors have recently rated.

#### Note

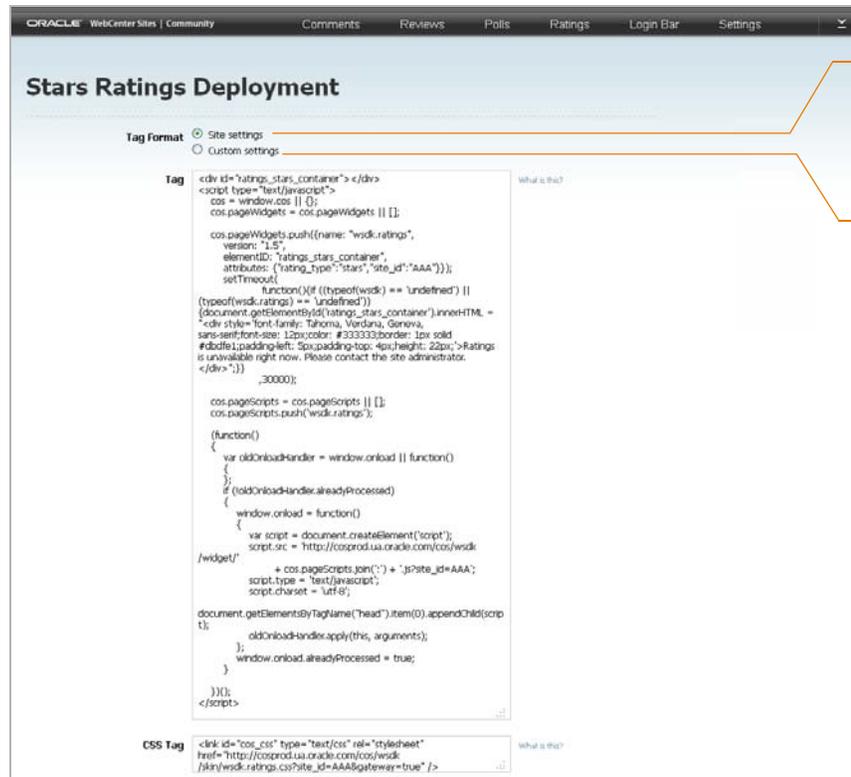
By default, the Recently Rated widget lists topics that have an associated ratings widget of resource type “Others.” To configure the Recently Rated widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Recently Rated widget tag). For more information, see “[Recently Rated Widget Tag](#),” on page 247

- **Most Rated** – to deploy the Most Rated widget tag. When this tag is deployed, it displays a list of the top ten web pages (topics) site visitors frequently rate.

#### Note

By default, the Most Rated widget lists topics that have an associated ratings widget of resource type “Others.” To configure the Most Rated widget to list topics of all resource types, you will have to modify the widget tag by removing the “resource\_type” : “other” parameter (in [line 8](#) of the Most Rated widget tag). For more information, see “[Most Rated Widget Tag](#),” on page 250

The deployment screen for each widget tag looks similar to the “Stars Ratings Deployment” screen, shown below:



**Site settings** is used to deploy the widget tag with its default settings.

**Custom settings** enables you to customize the tag directly in the tag generator. Because the settings are not saved, a deployed custom tag cannot be dynamically updated on the pages where it is displayed. It must be redeployed. For more information, see [“Deploying Custom Ratings Widget Tags,”](#) on page 120.

2. In the “Tag Format” field, make sure **Site settings** is selected.
3. Deploy the widget tag:
  - a. In the “Tag” field, copy (**Ctrl+C**) the widget tag.
  - b. Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For detailed instructions, see [steps 4 – 8 on page 29](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

## Deploying Custom Ratings Widget Tags

You can customize any widget tag related to rating functionality by setting its properties using the tag generator in the widget's "Deployment" screen. Each time a widget tag is customized, it must be re-deployed because custom values cannot be saved in the Community interface, instead they are saved only in the particular instance of the widget tag.

### To configure and deploy custom ratings widget tags

1. In the menu bar, select **Ratings > Deploy > name of the widget tag you wish to customize and deploy**:
  - **Stars Ratings** – to customize and deploy the Stars Ratings widget tag: (1) Configure the Stars Ratings widget tag with a unique identifier and resource type so that it can be referred to by other rating related widgets for its content, such as the number of ratings and average rating. (2) Specify the web page (topic title and URL) on which the Stars Ratings widget tag is deployed. (3) Specify whether a Login Bar widget will be displayed with the Stars Ratings widget.
  - **Thumbs Up/Down Ratings** – to customize and deploy the Thumbs Up/Down Ratings widget tag: that is, (1) Configure the Thumbs Up/Down Ratings widget tag with a unique identifier and resource type so that you can deploy multiple Thumbs Up/Down Ratings widget tags on the same page and so other rating related widgets can refer to its content, such as the date of the most recently posted rating. (2) Specify the web page (topic title and URL) on which the Thumbs Up/Down Ratings widget tag is deployed. (3) Specify whether a Login Bar widget will be displayed with the Thumbs Up/Down Ratings widget.
  - **Like It Ratings** – to customize and deploy the Like It Ratings widget tag: that is, (1) Configure the Like It Ratings widget tag with a unique identifier and resource type so that you can deploy multiple Like It Ratings widget tags on the same page and so other rating related widgets can refer to its content, such as the date of the most recently posted rating. (2) Specify the web page (topic title and URL) on which the Like It Ratings widget tag is deployed. (3) Specify whether a Login Bar widget will be displayed with the Like It Ratings widget.
  - **Recommend Ratings** – to customize and deploy the Recommend Ratings widget tag: that is, (1) Configure the Recommend Ratings widget tag with a unique identifier and resource type so that you can deploy multiple Recommend Ratings widgets on the same page and so other rating related widgets can refer to its content, such as the date of the most recently posted rating. (2) Specify the web page (topic title and URL) on which the Recommend Ratings widget tag is deployed. (3) Specify whether a Login Bar widget will be displayed with the Recommend Ratings widget.

- **Average Rating** – to customize and deploy the Ratings Average Rating widget tag: that is, for the Ratings Average Rating widget to display the average rating of a given topic (web page), the Ratings Average Rating widget must be configured to refer to the Stars Ratings widget, deployed on the desired topic (web page), by its resource ID.

#### Note

You can configure this widget to display the average rating of multiple web pages. For instructions, see [“Ratings Average Rating Widget Tag,” on page 243](#).

- **Recently Rated** – to customize and deploy the Recently Rated widget tag: that is, have the Recently Rated widget display recently rated topics of a desired type (instead of topics of type “Others”). For example, topics of type “Article” are posted on a web page; each article has its own ratings widget (Stars, Thumbs Up/Down, Like It, and/or Recommend) of resource type “Article.” If the Recently Rated widget is to list the recently rated topics of type “Article,” it must be configured to refer to the ratings widget(s) by resource type.

#### Note

To configure the Recently Rated widget to list topics of all resource types, you will have to modify the widget tag by removing the `“resource_type” : “other”` parameter (in [line 8](#) of the Recently Rated widget tag). For more information, see [“Recently Rated Widget Tag,” on page 247](#)

- **Most Rated** – to customize and deploy the Most Rated widget tag: that is, have the Most Rated widget display the most popular topics of a desired type (instead of topics of type “Others”). For example, topics of type “Article” are posted on a web page; each article has its own ratings widget (Stars, Thumbs Up/Down, Like It, and/or Recommend) of resource type “Article.” If the Most Rated widget is to list the most popular topics of type “Article,” it must be configured to refer to the ratings widget(s) by resource type.

#### Note

To configure the Most Rated widget to list topics of all resource types, you will have to modify the widget tag by removing the `“resource_type” : “other”` parameter (in [line 8](#) of the Most Rated widget tag). For more information, see [“Most Rated Widget Tag,” on page 250](#)

2. In the “Tag Format” field, select **Custom settings**. Customize the desired widget tag by setting values for the available fields in the tag’s custom deployment screen.

For field definitions, see:

- [Ratings > Deploy > Stars Ratings > Custom Settings](#)

- [Ratings > Deploy > Thumbs Up/Down Ratings > Custom Settings](#)
  - [Ratings > Deploy > Like It Ratings > Custom Settings](#)
  - [Ratings > Deploy > Recommend Ratings > Custom Settings](#)
  - [Ratings > Deploy > Average Rating](#)
  - [Ratings > Deploy > Recently Rated](#)
  - [Ratings > Deploy > Most Rated](#)
- 3.** Deploy the custom tag:
- a.** In the “Tag” field, copy (**Ctrl+C**) the widget tag.
  - b.** Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For detailed instructions, see [steps 4 – 8 on page 29](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

## Custom Ratings Widget Tag Deployment Screens

Once you access a given widget's custom deployment screen, you can use the available fields to set custom (temporary) values for the widget tag. These values are not saved, except in the template to which you copy the customized instance of the tag. This section provides field definitions.

### Ratings > Deploy > Stars Ratings > Custom Settings

This screen is used to define custom values for a particular instance of the Stars Ratings widget tag, which can then be deployed on web pages.

**Stars Ratings Deployment**

Tag Format:  Site settings  Custom settings

Resource ID:

Resource Type:

Resource Title:

Resource URL:

Login Bar Required:

Tag

```
<div id="ratings_stars_container_other"></div>
<script type="text/javascript">
  cos = window.cos || {};
  cos.pageWidgets = cos.pageWidgets || [];
  cos.pageWidgets.push({name: "wsdk.ratings",
    version: "1.5",
    elementId: "ratings_stars_container_other",
    attributes:
      {"rating_type": "stars", "resource_type": "other", "site_id": "AAA"});
  setTimeout(
    function(){if (typeof(wsdk) == "undefined" ||
      (typeof(wsdk.ratings) == "undefined"))
      {document.getElementById("ratings_stars_container_other").innerHTML
      += "<div style='font-family: Tahoma, Verdana, Geneva,
      sans-serif;font-size: 12px;color: #333333;border: 1px solid
      #d9d9d9;padding-left: 5px;padding-top: 4px;height: 22px;'>Ratings
      is unavailable right now. Please contact the site administrator.
      </div>";}
    },30000);
  cos.pageScripts = cos.pageScripts || [];
  cos.pageScripts.push("wsdk.ratings");
  (function()
  {
    var oldOnloadHandler = window.onload || function()
    {
      };
    if (!oldOnloadHandler.alreadyProcessed)
    {
      window.onload = function()
      {
        var script = document.createElement("script");
        script.src = "http://cosprod.ua.oracle.com/cos/wsdk
        /widget?
        + cos.pageScripts.join("&") + '&site_id=AAA';
        script.type = "text/javascript";
        script.charset = "utf-8";
        document.getElementsByTagName("head").item(0).appendChild(script);
        oldOnloadHandler.apply(this, arguments);
      };
      window.onload.alreadyProcessed = true;
    }
  })();
</script>
```

CSS Tag

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cosprod.ua.oracle.com/cos/wsdk
/skin/wsdk.ratings.css?site_id=AAA&gateway=true" />
```

### Note

The **Default** field value references the setting that is specified in the widget's "Appearance" screen. If you choose a default setting for your custom tag, and you deploy the tag, the default setting will be dynamically updated if its value is modified and saved in the Community interface.

**Table 32:** Customizing and Deploying the Stars Ratings Widget Tag

Field	Description
<b>Resource ID</b>	<p>Enter a resource ID (unique identifier) for this instance of the Stars Ratings widget tag. This resource ID can be used in other ratings widget tags (for example, the Ratings Average Rating widget tag) to refer to this instance of the Stars Ratings widget. You can enter a maximum of 50 English alphanumeric characters into this field.</p> <p><b>Note:</b> This field is required only if you want to configure other rating related widget tags to refer to this Stars Ratings widget's content (for example, the average rating of the web page).</p>
<b>Resource Type</b>	<p>Select a resource type (topic type) for this instance of the Stars Ratings widget tag. The resource type of a Stars Ratings widget tag categorizes the web page on which it is deployed. If you do not specify a resource type, the topic (web page) on which this instance of the Stars Ratings widget tag is deployed is associated with the default "Others" resource type.</p> <p><b>Note:</b> The current category file ("<a href="#">Ratings &gt; Configure &gt; Appearance</a>") determines the resource types that are available to you.</p>
<b>Resource Title</b>	<p>Enter the title of the web page on which this instance of the Stars Ratings widget tag will be deployed.</p>
<b>Resource URL</b>	<p>Enter the URL of the web page on which this instance of the Stars Ratings widget tag will be deployed.</p>
<b>Login Bar Required</b>	<p>Select either <b>Yes</b> or <b>No</b> to determine whether a Login Bar widget tag will be embedded in and deployed with this instance of the Stars Ratings widget tag.</p> <p><b>Note:</b> If you select <b>No</b>, you can still deploy a standalone Login Bar widget tag on the same page as the Stars Ratings widget (for instructions, see "<a href="#">Deploying a Login Bar Widget Tag</a>," on <a href="#">page 140</a>).</p>

## Ratings > Deploy > Thumbs Up/Down Ratings > Custom Settings

This screen is used to define custom values for a particular instance of the Thumbs Up/Down Ratings widget tag, which can then be deployed on web pages.

**Thumbs Up/Down Ratings Deployment**

Tag Format:  Site settings,  Custom settings

Resource ID:

Resource Type: Others

Resource Title:

Resource URL:

Login Bar Required: Default

**Tag**

```
<div id="ratings_thumbs_container_other"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || {};
cos.pageWidgets.push({name: "wsdk.ratings",
version: "1.5",
elementID: "ratings_thumbs_container_other",
attribute:
{"rating_type": "thumbs", "resource_type": "other", "site_id": "AAA"}});
setTimeout(
function(){if (typeof(wsdk) == 'undefined') ||
(typeof(wsdk.ratings) == 'undefined')
{document.getElementById("ratings_thumbs_container_other").innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva, sans-serif;font-size: 12px;color: #333333;border: 1px solid #808080;padding-left: 5px;padding-top: 4px;height: 22px;'>Ratings is unavailable right now. Please contact the site administrator.
</div>";}
},30000);
cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsdk.ratings");
(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://cosprod.us.oracle.com/cos/wsdk
/widget/
+ cos.pageScripts.join(";") + ".js?site_id=AAA";
script.type = "text/javascript";
script.charset = "utf8";
document.getElementsByTagName("head").item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
};
window.onload.alreadyProcessed = true;
}
})();
</script>
```

**CSS Tag**

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsdk
/widget/wsdk.ratings.css?site_id=AAA&geteway=true" />
```

### Note

The **Default** field value references the setting that is specified in the widget's "Appearance" screen. If you choose a default setting for your custom tag, and you deploy the tag, the default setting will be dynamically updated if its value is modified and saved in the Community interface.

**Table 33:** Customizing and Deploying the Thumbs Up/Down Ratings Widget Tag

Field	Description
<b>Resource ID</b>	<p>Enter a resource ID (unique identifier) for this instance of the Thumbs Up/Down Ratings widget tag. If you are deploying multiple Thumbs Up/Down Ratings widget tags on the same page, this resource ID is used to uniquely identify this instance of the Thumbs Up/Down Ratings widget tag. You can enter a maximum of 50 English alphanumeric characters into this field.</p> <p><b>Note:</b> This field is required only if you want to deploy multiple Thumbs Up/Down Ratings widget tags on the same page.</p>
<b>Resource Type</b>	<p>Select a resource type (topic type) for this instance of the Thumbs Up/Down Ratings widget tag. The resource type of a Thumbs Up/Down Ratings widget tag categorizes the web page on which it is deployed. If you do not specify a resource type, the topic (web page) on which this instance of the Thumbs Up/Down Ratings widget tag is deployed, is associated with the default “Others” resource type.</p> <p><b>Note:</b> The current category file (“<a href="#">Ratings &gt; Configure &gt; Appearance</a>”) determines the resource types that are available to you.</p>
<b>Resource Title</b>	<p>Enter the title of the web page on which this instance of the Thumbs Up/Down Ratings widget tag will be deployed.</p>
<b>Resource URL</b>	<p>Enter the URL of the web page on which this instance of the Thumbs Up/Down Ratings widget tag will be deployed.</p>
<b>Login Bar Required</b>	<p>Select either <b>Yes</b> or <b>No</b> to determine whether a Login Bar widget tag will be embedded in and deployed with this instance of the Thumbs Up/Down Ratings widget tag.</p> <p><b>Note:</b> If you select <b>No</b>, you can still deploy a standalone Login Bar widget tag on the same page as the Thumbs Up/Down Ratings widget (for instructions, see “<a href="#">Deploying a Login Bar Widget Tag</a>,” on page 140).</p>

## Ratings > Deploy > Like It Ratings > Custom Settings

This screen is used to define custom values for a particular instance of the Like It Ratings widget tag, which can then be deployed on web pages.

**Like It Ratings Deployment**

Tag Format:  Site settings  Custom settings

Resource ID:

Resource Type:

Resource Title:

Resource URL:

Login Bar Required:

**Tag**

```
<div id="ratings_like_it_container_other"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || [];
cos.pageWidgets.push({name: "wsdk.ratings",
version: "1.5",
elementId: "ratings_like_it_container_other",
attributes:
{rating_type:"like_it","resource_type":"other","site_id":"AAA"}});
setTimeout(
function(){if (typeof(wsdk) == 'undefined') ||
(typeof(wsdk.ratings) == 'undefined')
{document.getElementById("ratings_like_it_container_other").innerHTML = "<div style='font-family:Tahoma,Verdana,Geneva,sans-serif;font-size: 12px;color: #333333;border: 1px solid #000;padding-left: 5px;padding-top: 4px;height: 22px;'>Ratings is unavailable right now. Please contact the site administrator.
</div*");
},30000);
cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsdk.ratings");
(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "https://cosprod.ua.oracle.com/cos/wsdk/
/widget/
+ cos.pageScripts.join(";") + ".js?site_id=AAA";
script.type = "text/javascript";
script.charset = "utf8";
document.getElementsByTagName("head").item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
};
window.onload.alreadyProcessed = true;
}
})();
</script>
```

**CSS Tag**

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="https://cosprod.ua.oracle.com/cos/wsdk/
/js/wsdk.ratings.css?site_id=AAA&oteway=true" />
```

### Note

The **Default** field value references the setting that is specified in the widget's "Appearance" screen. If you choose a default setting for your custom tag, and you deploy the tag, the default setting will be dynamically updated if its value is modified and saved in the Community interface.

**Table 34:** Customizing and Deploying the Like It Ratings Widget Tag

Field	Description
<b>Resource ID</b>	<p>Enter a resource ID (unique identifier) for this instance of the Like It Ratings widget tag. If you are deploying multiple Like It Ratings widget tags on the same page, this resource ID is used to uniquely identify this instance of the Like It Ratings widget tag. You can enter a maximum of 50 English alphanumeric characters into this field.</p> <p><b>Note:</b> This field is required only if you want to deploy multiple Like It Ratings widget tags on the same page.</p>
<b>Resource Type</b>	<p>Select a resource type (topic type) for this instance of the Like It Ratings widget tag. The resource type of a Like It Ratings widget tag categorizes the web page on which it is deployed. If you do not specify a resource type, the topic (web page) on which this instance of the Like It Ratings widget tag is deployed is associated with the default “Others” resource type.</p> <p><b>Note:</b> The current category file (“<a href="#">Ratings &gt; Configure &gt; Appearance</a>”) determines the resource types that are available to you.</p>
<b>Resource Title</b>	<p>Enter the title of the web page on which this instance of the Like It Ratings widget tag will be deployed.</p>
<b>Resource URL</b>	<p>Enter the URL of the web page on which this instance of the Like It Ratings widget tag will be deployed.</p>
<b>Login Bar Required</b>	<p>Select either <b>Yes</b> or <b>No</b> to determine whether a Login Bar widget tag will be embedded in and deployed with this instance of the Like It Ratings widget tag.</p> <p><b>Note:</b> If you select <b>No</b>, you can still deploy a standalone Login Bar widget tag on the same page as the Like It Ratings widget tag (for instructions, see “<a href="#">Deploying a Login Bar Widget Tag</a>,” on <a href="#">page 140</a>).</p>

## Ratings > Deploy > Recommend Ratings > Custom Settings

This screen is used to define custom values for a particular instance of the Recommend Ratings widget tag, which can then be deployed on web pages.

**Recommend Ratings Deployment**

Tag Format:  Site settings,  Custom settings

Resource ID:

Resource Type:

Resource Title:

Resource URL:

Login Bar Required:

**Tag**

```
<div id="ratings_recommend_container_other"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || [];
cos.pageWidgets.push({name: "wsdk.ratings",
version: "1.5",
elementID: "ratings_recommend_container_other",
attribute:
{"rating_type": "recommend", "resource_type": "other", "site_id": "AAA"
}});
setTimeout(
function(){if (typeof(wsdk) == 'undefined') ||
(typeof(wsdk.ratings) == 'undefined')}
{document.getElementById("ratings_recommend_container_other").i
nnerHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
sans-serif;font-size: 12px;color: #808080;border: 1px solid
#d3d3d3;padding-left: 5px;padding-top: 4px;height: 22px;'>Ratings
is unavailable right now. Please contact the site administrator.
</div>"}
,30000);
cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push('wsdk.ratings');
(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement('script');
script.src = 'http://cosprod.us.oracle.com/cos/wsdk
/widget/'
+ cos.pageScripts.join(',') + '?site_id=AAA';
script.type = 'text/javascript';
script.charset = 'utf-8';
document.getElementsByTagName("head").item(0).appendChild(scrip
t);
oldOnloadHandler.apply(this, arguments);
};
window.onload.alreadyProcessed = true;
}
})();
</script>
```

**CSS Tag**

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsdk
/skin/wsdk.ratings.css?site_id=AAA&gateway=true" />
```

### Note

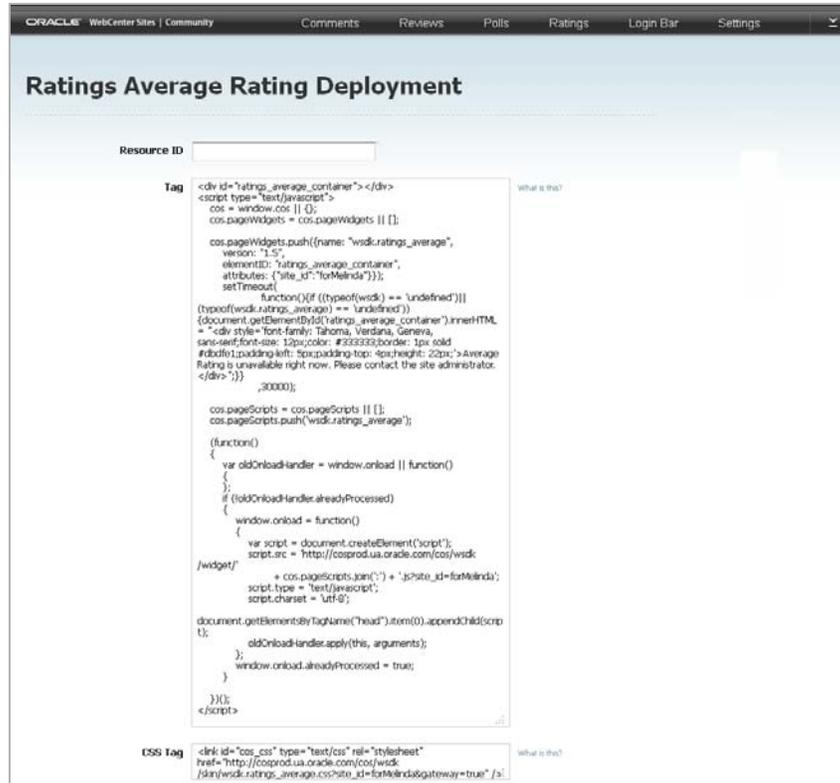
The **Default** field value references the setting that is specified in the widget's "Appearance" screen. If you choose a default setting for your custom tag, and you deploy the tag, the default setting will be dynamically updated if its value is modified and saved in the Community interface.

**Table 35:** Customizing and Deploying the Recommend Ratings Widget Tag

Field	Description
<b>Resource ID</b>	<p>Enter a resource ID (unique identifier) for this instance of the Recommend Ratings widget tag. If you are deploying multiple Recommend Ratings widget tags on the same page, this resource ID is used to uniquely identify this instance of the Recommend Ratings widget tag. You can enter a maximum of 50 English alphanumeric characters into this field.</p> <p><b>Note:</b> This field is required only if you want to deploy multiple Recommend Ratings widget tags on the same page.</p>
<b>Resource Type</b>	<p>Select a resource type (topic type) for this instance of the Recommend Ratings widget tag. The resource type of a Recommend Ratings widget tag categorizes the web page on which it is deployed. If you do not specify a resource type, the topic (web page) on which this instance of the Recommend Ratings widget tag is deployed is associated with the default “Others” resource type.</p> <p><b>Note:</b> The current category file (“<a href="#">Ratings &gt; Configure &gt; Appearance</a>”) determines the resource types that are available to you.</p>
<b>Resource Title</b>	<p>Enter the title of the web page on which this instance of the Recommend Ratings widget tag will be deployed.</p>
<b>Resource URL</b>	<p>Enter the URL of the web page on which this instance of the Recommend Ratings widget tag will be deployed.</p>
<b>Login Bar Required</b>	<p>Select either <b>Yes</b> or <b>No</b> to determine whether a Login Bar widget tag will be embedded in and deployed with this instance of the Recommend Ratings widget tag.</p> <p><b>Note:</b> If you select <b>No</b>, you can still deploy a standalone Login Bar widget tag on the same page as this instance of the Recommend Ratings widget (for instructions, see “<a href="#">Deploying a Login Bar Widget Tag,</a>” on page 140).</p>

## Ratings > Deploy > Average Rating

This screen is used to define custom values for a particular instance of the Ratings Average Rating widget tag, which can then be deployed on web pages.



**Table 36:** Customizing and Deploying the Average Rating Widget Tag

Field	Description
<b>Resource ID</b>	<p>Specify the resource ID (unique identifier) of a Stars Ratings widget. When this instance of the Ratings Average Rating widget is deployed it displays the average rating of the web page on which the specified Stars Ratings widget is deployed.</p> <p><b>Note:</b> To configure the Ratings Average Rating widget to display the average rating of multiple web pages on which Stars Ratings widgets are deployed, see <a href="#">“Ratings Average Rating Widget Tag,”</a> on <a href="#">page 243</a>.</p>

## Ratings > Deploy > Recently Rated

This screen is used to define custom values for a particular instance of the Recently Rated widget tag, which can then be deployed on web pages.

The screenshot shows the 'Recently Rated Deployment' configuration page. The 'Resource Type' dropdown is set to 'Others'. The 'Number of Topics' field contains the value '10'. The 'Include Since' field is empty. The 'Tag' field contains a complex JavaScript snippet for widget deployment. The 'CSS Tag' field contains a CSS link tag for the widget's stylesheet.

**Table 37:** Customizing and Deploying the Recently Rated Widget Tag

Field	Description
<b>Resource Type</b>	Select a resource type (topic type). When the Recently Rated widget is deployed, it lists the recently rated topics of a certain type (for example, “Article”) on which a ratings widget assigned to the specified resource type is deployed. If the deployed ratings widgets are not assigned a resource type, select <b>Others</b> . <b>Default Value: Others</b> <b>Note:</b> The current category file (“ <a href="#">Ratings &gt; Configure &gt; Appearance</a> ”) determines the resource types that are available to you.
<b>Number of Topics</b>	Specify the number of web pages (topics) to be included in this list (for example, enter 5 to display the five most recently rated topics). <b>Default Value: 10</b>

**Table 37:** Customizing and Deploying the Recently Rated Widget Tag *(continued)*

Field	Description
<b>Include Since</b>	<p>Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Recently Rated widget lists only the web pages whose ratings were posted on or after the specified date. This allows you to avoid displaying outdated pages and ratings.</p> <p><b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).</p>

## Ratings > Deploy > Most Rated

This screen is used to define custom values for a particular instance of the Most Rated widget tag, which can then be deployed on web pages.

The screenshot shows the 'Most Rated Deployment' configuration page. The 'Resource Type' dropdown is set to 'Others'. The 'Number of Topics' input field contains the value '10'. The 'Include Since' field is empty. The 'Tag' field contains the following JavaScript code:

```

<div id="most_rated_other_container"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || [];
cos.pageWidgets.push({names: "wsdk.topics",
version: "1.0",
elementID: "most_rated_other_container",
attributes:
{content_type: "most_rated", resource_type: "other", site_id: "forMelinda", count_topics: "10"}});
setTimeout(
function(){if ((typeof(wsdk) ==
undefined) || (typeof(wsdk.topics) == "undefined"))
{document.getElementById("most_rated_other_container").innerHTML
= "<div style='font-family: Tahoma, Verdana, Geneva,
serif;font-size: 12pt;color: #333333;border: 1px solid
#d3d3d3;padding-left: 5px;vertical-align: top;'> Topics is
unavailable right now. Please contact the site administrator.</div>";
},30000);
cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsdk.topics");
(function()
{
var oldOnloadHandler = window.onload || function()
{
};
if (!oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://coprod.us.oracle.com/cos/wsdk
/widget/
+ cos.pageScripts.join(";") + ".js?site_id=forMelinda";
script.type = "text/javascript";
script.charset = "utf-8";
document.getElementsByTagName("head").item(0).appendChild(script);
};
oldOnloadHandler.apply(this, arguments);
};
window.onload.alreadyProcessed = true;
})();
</script>

```

The 'CSS Tag' field contains the following CSS link tag:

```

<link id="cos_css" type="text/css" rel="stylesheet"
href="http://coprod.us.oracle.com/cos/wsdk
/skin/wsdk.topics.css?site_id=forMelinda&gateway=true" />

```

**Table 38:** Customizing and Deploying the Most Rated Widget Tag

Field	Description
Resource Type	<p>Select a resource type (topic type). When the Most Rated widget is deployed, it lists the most popular topics of a certain type (for example “Article”) on which a ratings widget assigned to the specified resource type is deployed. If the deployed ratings widgets are not assigned a resource type, select <b>Others</b>.</p> <p><b>Default Value:</b> Others</p> <p><b>Note:</b> The current category file (“<a href="#">Ratings &gt; Configure &gt; Appearance</a>”) determines the resource types that are available to you.</p>
Number of Topics	<p>Specify the number of web pages (topics) that can be included in this list (for example, enter 5 to display the five topics site visitors frequently rated).</p> <p><b>Default Value:</b> 10</p>

**Table 38:** Customizing and Deploying the Most Rated Widget Tag *(continued)*

Field	Description
<b>Include Since</b>	<p>Click in this field to render the date picker. Select a date by which to filter the web pages that will be listed. When deployed, the Most Rated widget lists only the web pages whose ratings were posted on or after the specified date. This allows you to avoid displaying outdated pages and ratings.</p> <p><b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).</p>



## Chapter 6

# Working with the Login Bar Widget

This chapter is for administrators and designers and provides a quick reference for reconfiguring settings for the Login Bar widget, such as specifying the types of visitor credentials (user name and password) that can be used to log in to the community widgets on your website (for example, Facebook credentials), and deploying a standalone Login Bar widget tag on a web page.

This chapter contains the following sections:

- [Overview](#)
- [Reconfiguring the Login Bar Widget's Default Settings](#)
- [Deploying a Login Bar Widget Tag](#)

## Overview



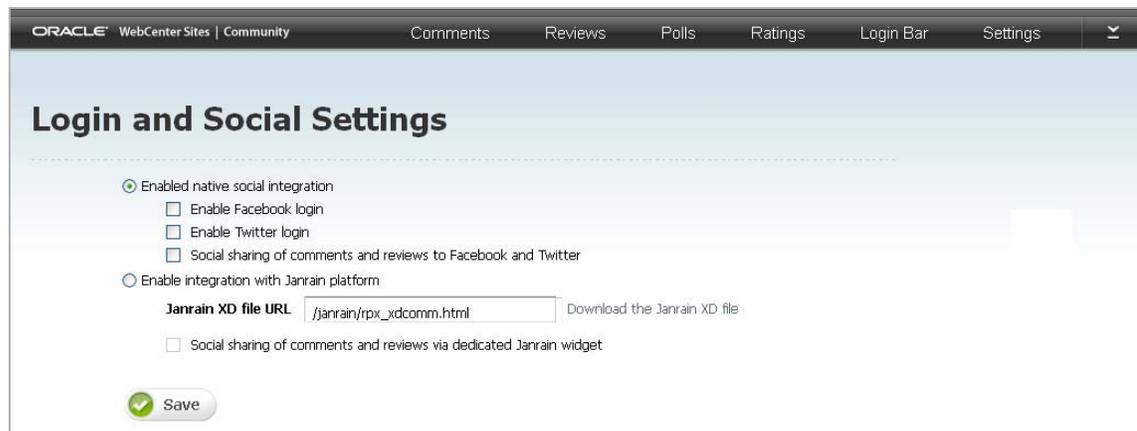
The **Login Bar** menu is used to reconfigure the login bar site visitors use to log in to the community widgets deployed on the pages of your company’s website. Configuring login settings includes determining which credentials (for example, Facebook and Twitter) visitors can use to log in to community widgets. Once you save these configurations, the system dynamically updates all deployed Login Bar widget tags (standalone and embedded) on pages where a login bar is displayed.

The instructions in this section are written as a quick reference and it is assumed you are already logged in to the Community interface. If you need instructions for logging in, see [“Logging In,” on page 26](#).

## Reconfiguring the Login Bar Widget’s Default Settings

The Login Bar widget’s default settings are its saved settings. These settings determine which credentials visitors can use to log in to the community widgets on your website, and whether visitors can share their comments and reviews with social network news feeds such as Facebook and Twitter.

To reconfigure the Login Bar widget’s default settings, administrators and designers can select **Login Bar > Configure** in the menu bar, which opens the “Login and Social Settings” screen (shown below). Once default settings are saved, they dynamically modify all Login Bar widget tags deployed on the website (including standalone instances and instances that are embedded in and deployed with other types of community widgets tags).



**Table 39:** Reconfiguring the Login Bar

Option	Description
<p><b>Enable native social integration</b></p>	<p>If your developers have registered the Community application with Facebook and/or Twitter, select <b>Enable native social integration</b>. This enables Facebook and Twitter to be used as external identity providers.</p> <p>Available only if the Login Bar widget is enabled for native social integration:</p> <ul style="list-style-type: none"> <li>• <b>Enable Facebook Login</b> allows visitors to log in to the community widgets on your website using their Facebook credentials.</li> <li>• <b>Enable Twitter Login</b> allows visitors to log in to the community widgets on your website using their Twitter credentials.</li> <li>• <b>Social sharing of comments and reviews to Facebook and Twitter</b> allows visitors to share the comments and reviews they post about the content of your website to their Facebook and Twitter news feeds.</li> </ul> <p><b>Note:</b> For information about registering the Community application with either of these external identity providers, see the <i>Oracle WebCenter Sites Developer's Guide for the Community Application</i>.</p>
<p><b>Enable integration with Janrain platform</b></p>	<p>Janrain is a third-party service for social login and social user profile management. If your developers have registered the Community application with the Janrain platform, and you want to allow visitors to log in to the community widgets on the website using their assigned credentials for any of the social networks supported by the Janrain platform, do the following:</p> <ol style="list-style-type: none"> <li>1. Select the <b>Enable integration with Janrain platform</b> radio button.</li> <li>2. In the “Janrain XD file URL” field, specify the exact location of the Janrain XD file for the domain on which you are deploying the widget tag.</li> <li>3. If you want to allow visitors to share posted comments and reviews to the social networks supported by Janrain, select <b>Social sharing of comments and reviews via dedicated Janrain widget</b>.</li> </ol> <p><b>Note:</b> If your developers have not registered the Community application with the Janrain platform, these options are greyed out (not available). For information about registering the Community application with the Janrain platform, see the <i>Oracle WebCenter Sites Developer's Guide for the Community Application</i>.</p>

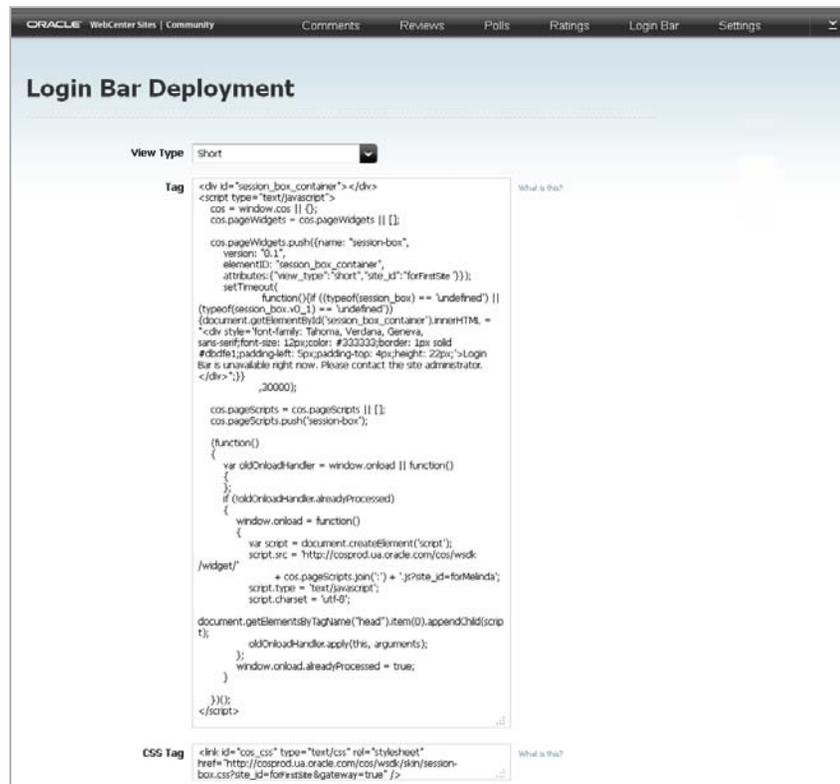
## Deploying a Login Bar Widget Tag

Administrators and designers can deploy the Login Bar widget tag with either default or custom settings by selecting **Login Bar > Deploy** in the menu bar. The Login Bar widget tag can be deployed as a standalone widget on a web page or embedded in the tag of another community widget. This section provides instructions for deploying a standalone Login Bar widget tag.

### To deploy a standalone login bar widget tag

1. In the menu bar, select **Login Bar > Deploy**.

The “Login Bar Deployment” screen opens:



2. If you want to customize this instance of the Login Bar widget tag, use the “View Type” field to select whether a **Short** or **Full** version of the Login Bar widget will be displayed on the website. Select one of the following options:
  - **Short** – When a visitor logs in to the website, the Login Bar widget displays the visitor’s user name as well as links which enable the visitor to either edit his profile or log out of the website. The website’s font styles are automatically applied to this version of the Login Bar widget.
  - **Full** – When a visitor logs in to the website, the Login Bar widget will display the visitor’s user name and profile picture, along with links that enable the visitor to edit his profile or log out of the website. This version of the Login Bar widget is displayed on the website with default font styles (which are specified in the Login Bar widget tag). The website’s font styles are not automatically applied to the Full version of the Login Bar widget.

**Note**

The value you select for the “View Type” field is considered a custom value. This value is not saved, except in the template to which you copy the Login Bar widget tag.

3. Deploy the widget tag:
  - a. In the “Tag” field, copy (**Ctrl+C**) the widget tag.
  - b. Insert the widget tag into the desired template, preview the asset to which the template is assigned, and publish the template to the website. For detailed instructions, see [steps 4 – 8 on page 29](#) in the section “[How Do I Deploy Community Widget Tags?](#)”



## Chapter 7

# Working with Polls

This chapter is for administrators and designers and provides a quick reference for creating and managing polls and the corresponding poll results.

This chapter contains the following sections:

- [Overview](#)
- [Working with the 'All Polls' Screen](#)
- [Creating a New Poll](#)

## Overview

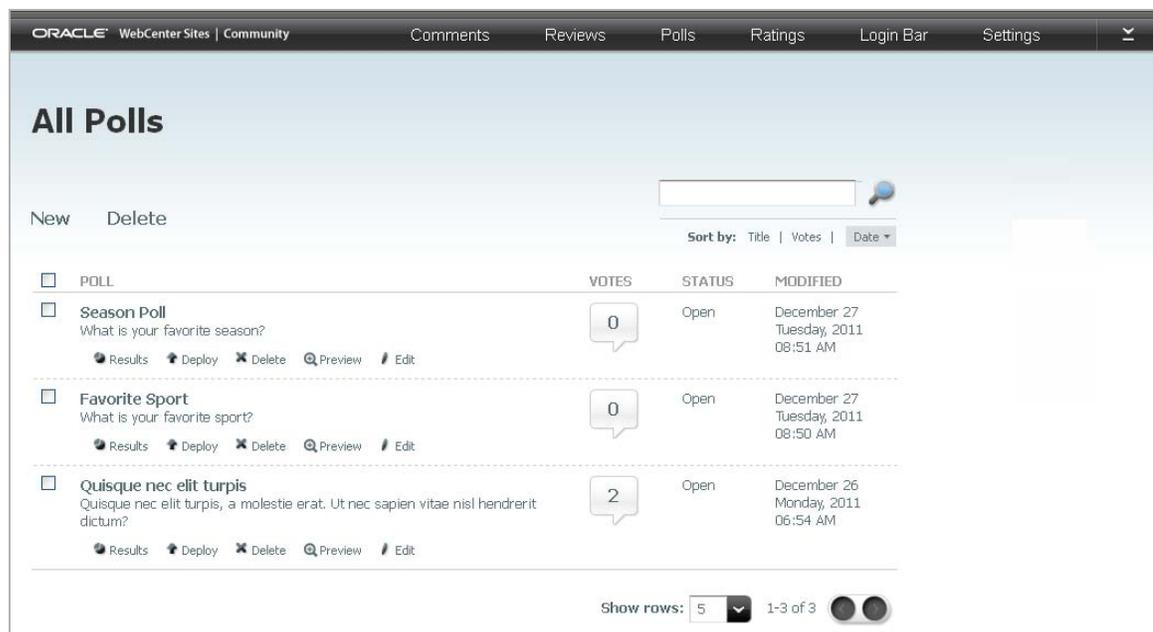


Polls allow you to survey the visitors of your website about a certain topic. The **Polls** menu is used to create Poll widgets and related Results widgets. The “Polls” menu provides you with options to moderate and reconfigure existing polls, and create new polls widgets which you can then deploy to a web page.

The instructions in this section are written as a quick reference and it is assumed you are already logged in to the Community interface. If you need instructions for logging in, see “[Logging In](#),” on page 26.

## Working with the ‘All Polls’ Screen

The “All Polls” screen (shown below) can be accessed by selecting **Polls > All Polls** in the menu bar. The “All Polls” screen is used to view existing polls and their results as well as to reconfigure, preview, and deploy poll widgets by using the quick action menu associated with each poll listed in the screen.



**Table 40:** Working with the “All Polls” screen

Action	Instructions
View the results of a deployed poll	<ol style="list-style-type: none"> <li>1. Navigate to the poll whose results you wish to view.</li> <li>2. In the poll’s quick action menu, click <b>Results</b>. A pop-up window opens displaying the results of the poll.</li> </ol>

**Table 40:** Working with the “All Polls” screen (continued)

Action	Instructions
<b>Deploy the polls widget tags</b>	<ol style="list-style-type: none"> <li>1. Navigate to the poll you wish to deploy.</li> <li>2. In the poll’s quick action menu, click <b>Deploy</b>. The “Deploy <i>Poll Name</i> Poll” screen opens. This screen allows you to deploy the following tags: <ul style="list-style-type: none"> <li>- <b>Poll Tag</b> – The main polls widget tag which contains all configured options for a particular poll. Deploying this tag on a web page displays the poll to visitors, who can then participate in voting. If the “Show Results” field is set to <b>Yes</b> (“<a href="#">Creating a New Poll</a>,” on page 146), the widget displays a summary of the poll’s results to visitors after they submit their vote for that poll.</li> <li>- <b>Results Tag</b> – The polls summary widget tag which displays only a summary of the main poll’s results. When this widget tag is deployed, the results it displays dynamically change with each poll vote.</li> </ul> </li> <li>3. Deploy the poll’s tag(s): <ol style="list-style-type: none"> <li>a) Click <b>Copy</b>.</li> <li>b) Insert the widget tag into the desired asset’s template, preview the asset, and publish the template to the website. For detailed instructions, see <a href="#">steps 4 – 8 on page 29</a> in the section “<a href="#">How Do I Deploy Community Widget Tags?</a>”</li> </ol> </li> </ol>
<b>Delete a poll</b>	<ol style="list-style-type: none"> <li>1. Navigate to the poll you wish to delete.</li> <li>2. In the poll’s quick actions menu, click <b>Delete</b>.</li> <li>3. In the confirmation box, click <b>OK</b>. The poll is removed from the Community interface. <b>Note:</b> If the poll you deleted is currently deployed on a page, the poll widget will display a message stating that the poll is currently unavailable. To remove the poll widget from the page, you must remove the poll’s widget tag from the template used to render that web page’s content.</li> </ol>
<b>Preview a poll</b>	<ol style="list-style-type: none"> <li>1. Navigate to the poll you wish to preview.</li> <li>2. In the poll’s quick action menu, click <b>Preview</b>. A pop-up window opens and displays a preview of the poll.</li> </ol>
<b>Modify an existing poll</b>	<ol style="list-style-type: none"> <li>1. Navigate to the poll you wish to modify.</li> <li>2. In the poll’s quick action menu, click <b>Edit</b>. The “Edit Poll” screen is displayed.</li> <li>3. Make the desired changes. For field definitions, see <a href="#">Table 41, on page 146</a>.</li> <li>4. Click <b>Save</b>. The modified poll is displayed in the “All Polls” screen and on the web page on which it is deployed.</li> </ol>

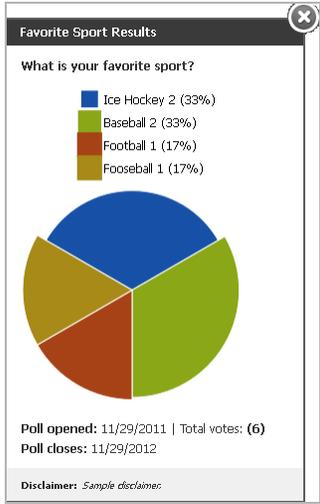
## Creating a New Poll

Administrators and designers can create a new poll by selecting **Polls > Create a New Poll** from the menu bar, which opens the “Create a New Poll” screen. Once you save the poll, it is listed in the “All Polls” screen (**Polls > All Polls**).

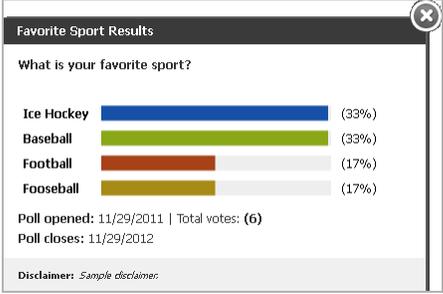
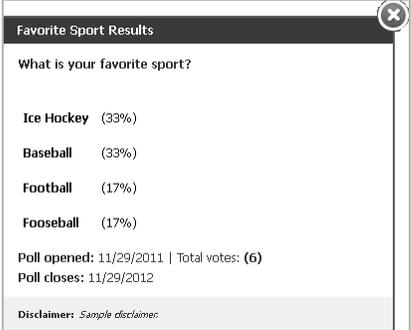
**Table 41:** Creating a Poll

Section	Field	Description
Questions/Answers	Poll Title	Enter a title for the poll.
	Poll Question	Enter the question you want to ask site visitors.

**Table 41:** Creating a Poll (continued)

Section	Field	Description
Questions/Answers (continued)	Answer <i>n</i>	<p>Enter the answer options from which you want visitors to choose.</p> <ul style="list-style-type: none"> <li>To include additional choices, click <b>Add Answer</b>.</li> <li>To remove an answer, click the <b>X</b> icon next to the answer you wish to remove.</li> <li>To change the color of an answer option when it is shown in the poll's results summary, enter the hex number of the desired color into the "Hex #" field, located next to the desired answer.</li> </ul>
Configure Results	Show Results	Select <b>Yes</b> to display a summary of the poll's results after a user submits an answer to the main poll. Select <b>No</b> to display only the main poll and its answer options.
	Graph Type	<p>If you selected <b>Yes</b> in the "Show Results" field, specify the type of graphical representation in which the poll's results summary will be displayed. Select one of the following:</p> <ul style="list-style-type: none"> <li><b>Pie Chart</b> – Displays the poll results in a pie chart format:</li> </ul> 

**Table 41:** Creating a Poll *(continued)*

Section	Field	Description
<b>Configure Results</b> <i>(continued)</i>	Graph Type <i>(continued)</i>	<ul style="list-style-type: none"> <li> <b>Bar Chart</b> – Displays the poll results in a bar chart format:            </li> <li> <b>Flat Results</b> – Displays the poll results in a list, as flat text without any graphics. The percentage of visitors who voted for each answer is displayed next to the corresponding answer:            </li> </ul>
	Results View	Specify whether poll results will be displayed in a pop-up window ( <b>Popup</b> ) or alongside the poll widget ( <b>In-Site</b> ). <b>Note:</b> This field is available only if you selected <b>Yes</b> in the “Show Results” field.
	Results Popup Width, px	Specify the width of the results pop-up window. The width must be at least 300 px. <b>Note:</b> This field is available only if you selected <b>Yes</b> in the “Show Results” field and <b>Popup</b> in the “Results View” field.
	Thank You Note	If you selected <b>Yes</b> in the “Show Results” field, enter a thank you note that will be displayed to site visitors when they participate in the poll.

**Table 41:** Creating a Poll *(continued)*

Section	Field	Description
<b>Configure Results</b> <i>(continued)</i>	Include Disclaimer	Select either <b>Yes</b> or <b>No</b> to specify whether a disclaimer will be displayed to visitors. <b>Note:</b> This field is available only if you selected <b>Yes</b> in the “Show Results” field.
	Disclaimer	If you selected <b>Yes</b> in the “Include Disclaimer” field, enter the disclaimer text you wish to display to visitors. When the polls widget is deployed, the disclaimer is displayed beneath the poll’s question and answers. <b>Note:</b> This field supports plain text and basic HTML tags. You can use HTML tags to control the appearance of your disclaimer
<b>Common</b>	Poll Status	This field allows you to manually open and close the poll. Select either: <ul style="list-style-type: none"> <li>• <b>Open</b> – visitors can participate in the poll.</li> <li>• <b>Closed</b> – visitors cannot submit votes.</li> </ul>
	Poll Closure Date	Click in this field to render the date picker. Specify the date the poll will close (votes can no longer be submitted). If you do not wish to specify a closure date for the poll, leave this field blank. <b>Note:</b> If you do not wish to use the date picker, you can enter a date using the format <i>dd-mm-yyyy</i> (for example, 02-04-2011).
	Theme	Specify the appearance of the polls widget as it will be displayed on the web page. You can select one of the following: <ul style="list-style-type: none"> <li>• <b>Basic</b> – Displays the poll title at the top of the widget. The question is displayed below the title and the available answers are displayed below the question.</li> </ul> 

**Table 41:** Creating a Poll *(continued)*

Section	Field	Description
<b>Common</b> <i>(continued)</i>	Theme <i>(continued)</i>	<ul style="list-style-type: none"> <li> <b>Advanced</b> – Displays the poll title in a grey two-toned title bar at the top of the widget. The question is displayed below the title in a blue background, and the available answers are displayed below the question.           </li> </ul>  <ul style="list-style-type: none"> <li> <b>No Design</b> – Displays only the poll question and its available answers, without a frame or border around them.           </li> </ul>  <ul style="list-style-type: none"> <li> <b>Open Design</b> – Displays the poll question and its available answers in an open source style design.           </li> </ul> 

## Chapter 8

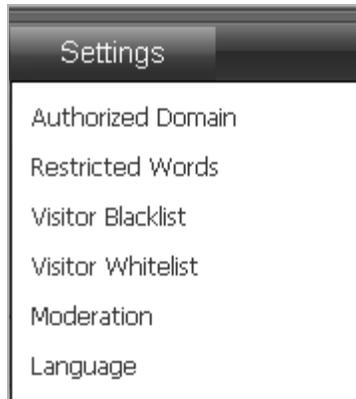
# Reconfiguring General Community Widget Settings

This chapter provides a quick reference for reconfiguring general settings that apply to all community widgets (to which the settings are applicable) of a given site.

This chapter contains the following sections:

- [Overview](#)
- [Working with the 'Settings' Menu](#)

## Overview



The **Settings** menu is used to configure moderation and security settings for the entire site and all widget tags. Configuring general community widget settings includes applying automatic moderation filters (“Restricted Words,” “Visitor Blacklist,” and “Visitor Whitelist”) to assist in the moderation process and protecting your widget tags from unauthorized deployment. Once you save these settings, the system dynamically updates all widget tags in the Community interface and on pages where tags are deployed.

The instructions in this section are written as a quick reference and it is assumed you are already logged in to the Community interface. If you need instructions for logging in, see “[Logging In](#),” on page 26.

## Working with the ‘Settings’ Menu

This section is for all Community application users. You can reconfigure general settings for community widgets (to which the settings are applicable) in the following ways:

### Settings > Authorized Domain

This screen can be accessed by administrators and designers and is used to protect all deployed tags from unauthorized use. Authorizing a domain enables you to specify the domain on which community widget tags can be deployed. This restricts the usage of the tags to only the specified domain.

A screenshot of the "Authorized Domain" settings page in a web application. The page has a dark grey header with the Oracle logo and navigation links: "WebCenter Sites | Community", "Comments", "Reviews", "Polls", "Ratings", "Login Bar", and "Settings". The main content area has a light blue background and is titled "Authorized Domain". Below the title, there is a text input field labeled "Domain URL". At the bottom of the form, there is a green "Save" button with a checkmark icon.

**Table 42:** Authorizing a Domain

Field	Description
<b>Domain URL</b>	<p>Enter the domain name of the site(s) on which you wish to display community widgets.</p> <p>For example:</p> <p><i>www.mycompany.com</i> or <i>www.mysite.mycompany.com</i></p> <p>To specify the domain name of multiple sites, separate each domain name with a comma. For example:</p> <p><i>www.mycompany.com, www.mycompany2.com, www.mycompany3.com, *.mycompany.com</i></p> <p>Widgets can be rendered only on the specified domain(s) and sub-domain(s). If a visitor views the source code of your company's site, and attempts to copy and deploy the widget tags to his own site, the deployment will fail.</p> <p><b>Note:</b> This field is optional. If you do not wish to specify a domain on which widget tags can be deployed, leave this field blank.</p>

## Settings > Restricted Words

This screen can be accessed by administrators and moderators. This screen is used to upload a word filter file, which will prevent the publication of comments and reviews containing words that are listed in the file. The comments and reviews will be marked as **Inappropriate** and will await approval by an administrator or moderator in the Community interface.

### Enabling the Restricted Word Filter

The Community application provides a default word filter that searches for partial matches to the words specified in the uploaded word filter file. To enable the word filter to search for these words, you will have to enable auto moderation against restricted words (“[Settings > Moderation](#),” on page 158).

**Table 43:** Configuring Restricted Words

Field	Description
<b>Upload Filter</b>	<p>Click <b>Browse</b> to upload a filter file. The size of the file you are uploading can be no more than 100KB, the file must be in .txt format, and each word contained in the file must be on its own line.</p> <p><b>Note:</b> When you upload a filter file and save it to the Community interface, the file is renamed <code>word_filter.txt</code>.</p>

## Settings > Visitor Blacklist

This screen can be accessed by administrators and moderators. This screen is used to prevent certain visitors' comments, reviews, and ratings from being displayed on the website without going through the moderation process. Administrators and moderators must manually moderate all comments, reviews, and ratings posted by blacklisted visitors.

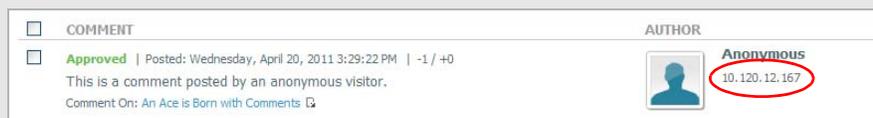


### Blacklisting Anonymous Visitors

If you are blacklisting anonymous visitors, you will need their IP addresses. A visitor's IP address is located next to each of his comments, reviews, and ratings listed in the Community interface.

#### To obtain a visitor's IP address

1. Access either the "All Comments," "All Reviews," or "All Ratings" screen.
2. Navigate to the comment, review, or rating posted by the anonymous visitor you wish to add to the blacklist. The visitor's IP address is listed under the "Author" column, next to his comment, review, or rating.

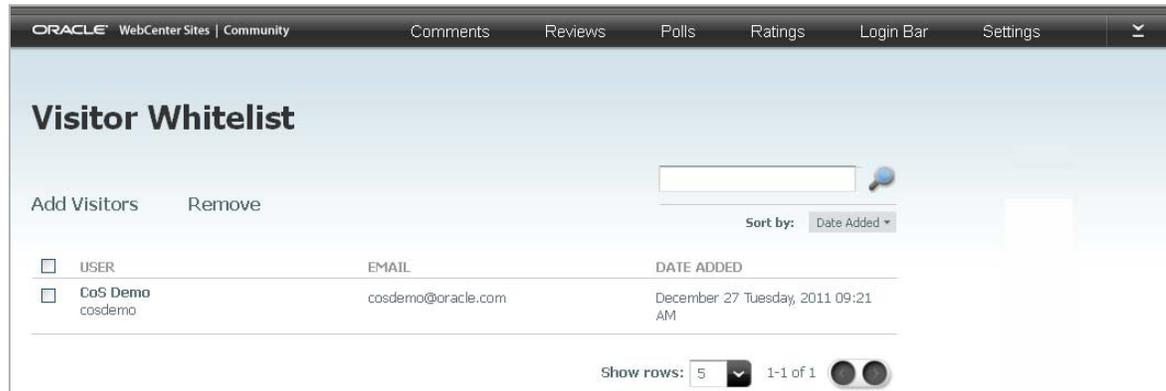


**Table 44:** Configuring the Visitor Blacklist

Action	Instructions
<b>Adding authenticated visitors to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>1. Click the <b>Add Visitors</b> link. The “Add Visitors to Blacklist” screen opens.</li> <li>2. In the <b>Available</b> list box, select the names of the visitors you wish to blacklist and move them to the <b>Selected</b> list box. (To search for a specific visitor, type his name in the <b>Filter List</b> field. The results appear in the <b>Available</b> list box.)</li> <li>3. (Optional) In the <b>Reason</b> field, enter the reason for blacklisting the visitors.</li> <li>4. Click <b>Save</b>. The visitors’ names are now listed in the “Visitor Blacklist” screen.</li> </ol>
<b>Adding an anonymous visitor to the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>1. Click the <b>Add IP Address</b> link. The “Add IP Address” screen opens.</li> <li>2. Fill in the following fields: <ul style="list-style-type: none"> <li>- <b>IP</b> – Enter the visitor’s IP address. For instruction, see “Blacklisting Anonymous Visitors” on <a href="#">page 155</a>.</li> <li>- (Optional) <b>Reason</b> – Enter the reason for blacklisting the visitor.</li> </ul> </li> <li>3. Click <b>Save</b>. The visitor’s IP address is now listed in the “Visitor Blacklist” screen.</li> </ol>
<b>Removing visitors from the “Visitor Blacklist”</b>	<ol style="list-style-type: none"> <li>1. Select the checkboxes of the visitors you wish to remove from the blacklist.</li> <li>2. Click the <b>Remove</b> link. The visitors’ names are removed from the “Visitor Blacklist” screen.</li> </ol>

## Settings > Visitor Whitelist

This screen can be accessed by administrators and moderators. This screen is used to allow authenticated visitors to post comments, reviews, and ratings without requiring approval. This means that once a visitor is added to the whitelist, all of the visitor's posts are automatically approved and displayed on the website (even if manual moderation is enabled).



### Note

Only authenticated visitors can be added to the visitor whitelist.

**Table 45:** Configuring the Visitor Whitelist

Action	Instructions
<b>Adding authenticated visitors to the "Visitor Whitelist"</b>	<ol style="list-style-type: none"> <li>1. Click the <b>Add Visitors</b> link. The "Add Visitors to Whitelist" screen opens.</li> <li>2. In the <b>Available</b> list box, select the names of the visitors you wish to add to the whitelist and move them to the <b>Selected</b> list box. (To search for a specific visitor, type his name in the <b>Filter List</b> field. The results appear in the <b>Available</b> list box.)</li> <li>3. Click <b>Save</b>. The visitors' names are now listed in the "Visitor Whitelist" screen.</li> </ol>
<b>Removing visitors from the "Visitor Whitelist"</b>	<ol style="list-style-type: none"> <li>1. Select the checkboxes of the visitors you wish to remove from the whitelist.</li> <li>2. Click the <b>Remove</b> link. The visitors' names are removed from the "Visitor Whitelist" screen.</li> </ol>

## Settings > Moderation

This screen can be accessed by administrators and moderators. This screen is used to specify the following options:

- Enable manual or automatic moderation. Allow comments, reviews, and ratings to be either passed through the moderation process or automatically approved and displayed on the website.
- Apply the auto-moderation filters to assist in the moderation process. The filters for auto-moderation are “Restricted Words,” “Visitor Blacklist,” and “Visitor Whitelist.” For information, see “[Settings > Restricted Words](#),” on page 154, “[Settings > Visitor Blacklist](#),” on page 155, and “[Settings > Visitor Whitelist](#),” on page 157.
- Choose to auto-moderate reviews and ratings by the rating a site visitor gives to a topic.
- Specify whether visitors can assist in the moderation process by allowing them to flag comments and reviews posted to the pages of the website. Also, specify which visitors (authenticated or all) can flag comments and reviews.

**Table 46:** Configuring Moderation Settings for Widgets

Field	Description
<b>Manual Moderation</b>	Enables you to control whether comments, reviews, and ratings will be automatically approved or manually moderated before being approved. You can also choose to enable the “Visitor Whitelist” filter. When applied, it approves all comments, reviews, and ratings posted by whitelisted visitors.

**Table 46:** Configuring Moderation Settings for Widgets *(continued)*

Field	Description
<b>Auto-Moderation</b>	Enables you to apply the “Restricted Words” and/or “Visitor Blacklist” filters. When these filters are applied, they mark as <b>Inappropriate</b> all comments and reviews that contain prohibited words (“Restricted Words”) and all comments, reviews, and ratings that are posted by blacklisted visitors (“Visitor Blacklist”).  You can also choose to automatically moderate reviews and ratings by the rating the site visitor who posted them gives to the topic. When enabled, any review or rating posted with a rating of 2 stars or less is automatically marked as <b>Inappropriate</b> .
<b>Flagging By Visitors</b>	Enables you to control site visitor flagging capabilities for approved comments and reviews.

## Settings > Language

This screen can be accessed by administrators and designers. This screen enables you to specify the language in which deployed community widgets will be displayed.

The screenshot shows the Oracle WebCenter Sites interface. At the top, there is a navigation bar with links for 'Comments', 'Reviews', 'Polls', 'Ratings', 'Login Bar', and 'Settings'. The main content area is titled 'Language'. It features a dropdown menu labeled 'Language' with 'English' selected. Below the dropdown is a 'Save' button with a green checkmark icon.

Field	Description
Language	<p>Click the down-arrow to display the “Language” drop-down menu. Select the language in which all deployed community widgets will be displayed.</p> <p>The available options are:</p> <ul style="list-style-type: none"> <li>• <b>English</b></li> <li>• <b>Brazilian Portuguese</b></li> <li>• <b>Simplified Chinese</b></li> <li>• <b>Traditional Chinese</b></li> <li>• <b>French</b></li> <li>• <b>German</b></li> <li>• <b>Italian</b></li> <li>• <b>Japanese</b></li> <li>• <b>Korean</b></li> <li>• <b>Spanish</b></li> </ul> <p><b>Note:</b> If the desired language is not listed, contact your WebCenter Sites administrator or developers. For instructions about adding language options to this list, see the <i>Oracle WebCenter Sites Developer’s Guide for the Community Application</i>.</p>

## Chapter 9

# Working with the Cache Tool

This chapter is for administrators and provides information about using the cache tool to view the information stored in the production and management caches and to verify that the caches contain valid information.

This chapter contains the following sections:

- [Overview](#)
- [Viewing Cache Information](#)

## Overview

All community widget data is stored in and retrieved from the WebCenter Sites database. This includes user generated content (for example, comments, reviews, and user profiles), and widget settings (for example, permission, appearance, and moderation settings).

When data is requested for the first time, it is read from the WebCenter Sites database and stored in either the management or production cache. A cache is a temporary storage area where data can be accessed quickly. Upon subsequent requests, the data is read directly from the cache in which it is stored (reducing the load on the database). The **Cache** tool is used to view information about the data stored in the caches and to validate the data.

## Viewing Cache Information

The first screen you see when you access the Cache tool is the cache summary screen for either the management or production system. The summary screen contains statistics about the data stored on the management and production systems. Detailed information about the data contained within the caches on each system can be viewed by clicking the name of the desired cache in the summary screen.

### To view cache information

1. Access the Cache tool by doing one of the following:
  - To view information about the data stored in the management caches, enter `http://<management:port>/cos/cache`  
where `<management:port>` is the host name and port number of the server running the management system.
  - To view information about the data stored in the production caches, enter `http://<production:port>/cos/cache`  
where `<production:port>` is the host name and port number of the server running the production system.
2. In the Login screen, log in to the Cache tool with your assigned credentials.  
In this guide we are using the default login credentials of a general administrator:  
**Username:** fwadmin  
**Password:** xceladmin
3. Click **Login**.

You will see one of the following cache summary screens:

<input type="checkbox"/>	Cache Name	Total Count	Memory Store Count	Disk Store Count	Cache Hits	In Memory Hits	On Disk Hits	Cache Misses	Average Get Time, ms	Max Elements In Memory
<input type="checkbox"/>	Common	123	123	0	260	260	0	188	108	100000
<input type="checkbox"/>	Dependencies	92	92	0	1186	1186	0	161	17	100000
<input type="checkbox"/>	Shared	7	7	0	85	85	0	60	282	100000

<input type="checkbox"/>	Cache Name	Total Count	Memory Store Count	Disk Store Count	Cache Hits	In Memory Hits	On Disk Hits	Cache Misses	Average Get Time, ms	Max Elements In Memory
<input type="checkbox"/>	Common	233	233	0	719	719	0	393	114	100000
<input type="checkbox"/>	Dependencies	142	142	0	2783	2783	0	184	150	100000
<input type="checkbox"/>	Shared	15	15	0	156	156	0	238	32	100000

The “Summary Management” screen contains statistics about the Community data stored in the management cache. This includes user generated content, widget settings, and any other data displayed in the Community interface. The “Summary Production” screen contains statistics about the Community data stored in the production cache. This includes user generated content, visitor profile information, and any other data displayed on the website.

[Table 47](#) provides information about the columns in the summary screen. The summary screen also provides the option to flush the caches on the management (or production) system. To flush the data stored in the caches, select the desired cache and click **Flush Cache**, located at the top of the summary screen.

**Table 47:** Overview of Columns in the Summary Screen

Column	Description
Cache Name	The name of the cache where Community application data is stored. The available caches are: <ul style="list-style-type: none"> <li>• Common</li> <li>• Dependencies</li> <li>• Shared</li> </ul> Click the desired cache name to display detailed information about the data stored in that cache.
Total Count	The number of objects currently stored in the cache.
Memory Store Count	The number of objects stored to memory.
Disk Store Count	The number of objects stored on disk.
Cache Hits	The total number of requested objects served from cache.
In Memory Hits	The total number of requested objects served from memory.
On Disk Hits	The total number of requested objects served from the disk.
Cache Misses	The total number of requested objects that were not stored in cache and so had to be read from the WebCenter Sites database.
Average Get Time, ms	The average time (in milliseconds) it took the Community application to retrieve requested data.
Max Elements In Memory	The maximum number of items allowed in the cache.

- View detailed information about the data contained within the caches on each system. In the “Cache Name” column of the summary screen, click the name of the desired cache. For more information, see:
  - [Viewing Information in the Common Cache](#)
  - [Viewing Information in the Dependencies Cache](#)
  - [Viewing Information in the Shared Cache](#)

## Viewing Information in the Common Cache

To access the “Common Cache” screen, click **Common** in the “Cache Name” column of the summary screen. The “Common Cache” screen will look similar to the following:

The common cache for the management system contains results of data queries necessary for displaying the Community interface to Community application users (that is, users assigned the roles `SiteAdmin`, `Moderator`, and `Designer`). The common cache for the production system contains results of data queries necessary for displaying community widgets on the website for site visitors. [Table 48](#) provides information about the columns in the “Common Cache” screen.

<input type="checkbox"/>	Title	Hit Count	Created Time	Last Access Time	Dependencies
<input type="checkbox"/>	wsdk.ui.user_service/register/register_view.shtml Key: wsdk.ui.user_service/register/register_view.shtml	2	Tuesday, December 27, 2011 11:59:13 AM	Tuesday, December 27, 2011 12:00:54 PM	
<input type="checkbox"/>	wsdk.ui.user_service/edit_profile/edit_profile_view.shtml Key: wsdk.ui.user_service/edit_profile/edit_profile_view.shtml	2	Tuesday, December 27, 2011 11:59:14 AM	Tuesday, December 27, 2011 12:00:54 PM	
<input type="checkbox"/>	wsdk.ui.thumbs/thumbs_view.shtml Key: wsdk.ui.thumbs/thumbs_view.shtml	2	Tuesday, December 27, 2011 11:59:13 AM	Tuesday, December 27, 2011 12:00:54 PM	
<input type="checkbox"/>	Search: User Link (tr3_user_link) Key: query_http://csoracle.ua.oracle.com:8080/cs/REST/sites/AdminSite/types/tr3_user_link/search?searchengine=dbsic&field:cos_username>equals=codemoc&count=2048&field:status>equals=PL&field:status>equals=ED	30	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:00:44 PM	unknowndeps tr3_user_link
<input type="checkbox"/>	Count: Rating (tr3_rating_record) Key: query_count_http://csoracle.ua.oracle.com:8080/cs/REST/sites/forMelinda/types/tr3_rating_record/search?searchengine=dbsic&field:cos_root_id>equals=1324666295068&field:cos_owner>equals=-1&field:cos_guest_id>equals=a2c8b8ea10b61af69aab3d2efaf451d8&field:cos_rating_type>equals=1&countonly=true	0	Tuesday, December 27, 2011 12:00:57 PM	Tuesday, December 27, 2011 12:00:57 PM	unknowndeps tr3_rating_record

**Table 48:** Overview of Columns in the Common Cache Screen

Column	Description
Title	Uniquely identifies each item in the common cache.
Hit Count	The total number of requested objects that were served by reading the common cache.
Created Time	The date the data was fetched from the WebCenter Sites database and then stored in the cache for future access.

**Table 48:** Overview of Columns in the Common Cache Screen

Column	Description
Last Access Time	The most recent date the data was retrieved from the common cache.
Dependencies	Lists the assets in the WebCenter Sites database on which the corresponding data depends. If the data is associated with more than one database table, multiple dependencies will be listed for that entry.

## Viewing Information in the Dependencies Cache

To access the “Dependencies Cache” screen, click **Dependencies** in the “Cache Name” column of the summary screen. The “Dependencies Cache” screen will look similar to the following:

<input type="checkbox"/>	Key	Hit Count	Generation Count	Created Time	Last Access Time
<input type="checkbox"/>	asset-1324666249611:tr3_st_value	19	0	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666249536:tr3_st_value	19	0	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666249461:tr3_st_value	19	0	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666249386:tr3_st_value	19	0	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666249311:tr3_st_value	19	0	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246925:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246850:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246700:tr3_st_value	8	0	Tuesday, December 27, 2011 11:59:19 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246625:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246550:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246475:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246400:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246325:tr3_st_value	4	0	Tuesday, December 27, 2011 11:59:20 AM	Tuesday, December 27, 2011 12:04:11 PM
<input type="checkbox"/>	asset-1324666246250:tr3_st_locator	1	0	Tuesday, December 27, 2011 11:59:19 AM	Tuesday, December 27, 2011 12:04:11 PM

Show rows: 100 101-142 of 142

All links to data in the database tables are stored in the dependencies cache. The “Dependencies Cache” screen lists query dependencies that exist on the management (or production) system. [Table 49](#) provides information about the columns in this screen.

**Table 49:** Overview of Columns in the Dependencies Cache screen

Column	Description
Key	Uniquely identifies each item in the dependencies cache.

**Table 49:** Overview of Columns in the Dependencies Cache screen *(continued)*

Column	Description
Hit Count	The total number of requested objects that were served by reading the dependencies cache.
Generation Count	The total number of changes made in the database to the corresponding entry since the application has been started.
Created time	The date the object was fetched from the WebCenter Sites database and then stored in the cache for future access.
Last Access Time	The most recent date the data was retrieved from the dependencies cache.

## Viewing Information in the Shared Cache

To access the “Shared Cache” screen, click **Shared** in the “Cache Name” column of the summary screen. The “Shared Cache” screen will look similar to the following:

<input type="checkbox"/>	Title	Hit Count	Created Time	Last Access Time	Dependencies
<input type="checkbox"/>	feedid_review_feedrevthumbs Key: feedid_review_feedrevthumbs	3	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:00:43 PM	
<input type="checkbox"/>	feedid_review_feedrevdeff Key: feedid_review_feedrevdeff	11	Tuesday, December 27, 2011 11:59:35 AM	Tuesday, December 27, 2011 12:00:43 PM	
<input type="checkbox"/>	feedid_review_feedThumbsUpDownReviewsDeployme Key: feedid_review_feedThumbsUpDownReviewsDeployment	0	Tuesday, December 27, 2011 11:27:42 AM	Tuesday, December 27, 2011 11:27:42 AM	
<input type="checkbox"/>	feedid_review_feedStarsReviewsDeployment Key: feedid_review_feedStarsReviewsDeployment	0	Tuesday, December 27, 2011 11:27:42 AM	Tuesday, December 27, 2011 11:27:42 AM	
<input type="checkbox"/>	feedid_rating_feedThumbsUpDown Key: feedid_rating_feedThumbsUpDown	0	Tuesday, December 27, 2011 11:27:44 AM	Tuesday, December 27, 2011 11:27:44 AM	
<input type="checkbox"/>	feedid_rating_feedStarsRatingsDeployment Key: feedid_rating_feedStarsRatingsDeployment	0	Tuesday, December 27, 2011 11:27:44 AM	Tuesday, December 27, 2011 11:27:44 AM	
<input type="checkbox"/>	feedid_rating_feedStarsR Key: feedid_rating_feedStarsR	0	Tuesday, December 27, 2011 12:00:57 PM	Tuesday, December 27, 2011 12:00:57 PM	
<input type="checkbox"/>	feedid_rating_feedRecommendRatingsDeployment Key: feedid_rating_feedRecommendRatingsDeployment	0	Tuesday, December 27, 2011 11:27:44 AM	Tuesday, December 27, 2011 11:27:44 AM	
<input type="checkbox"/>	SSO-ST-1143-hxnlQwqft64NnaGb2Yr-cas-csoracle.ua.oracle.com-1 Key: SSO-ST-1143-hxnlQwqft64NnaGb2Yr-cas-csoracle.ua.oracle.com-1	66	Tuesday, December 27, 2011 1:18:04 PM	Tuesday, December 27, 2011 1:37:03 PM	

Summary >

**Shared**

Delete

Sort by: Key ▾

Show rows: 100 ▾ 1-15 of 15

The Shared cache stores data that is shared among cluster members and enables the synchronization of cluster members. The data stored in the Shared cache includes feed identities, site identities, captcha verification data, single sign-on ticket states, restricted words, and authorized domains. The “Shared Cache” screen enables you to view the data stored in the shared cache. [Table 50](#) provides information about the columns in the “Shared Cache” screen.

**Table 50:** Overview of Columns in the Shared Cache screen

Column	Description
Title	Uniquely identifies each item in the shared cache.

**Table 50:** Overview of Columns in the Shared Cache screen *(continued)*

Column	Description
Hit Count	The total number of requested objects that were served by reading the shared cache.
Created time	The date the object was fetched from the WebCenter Sites database and then stored in the cache for future access.
Last Access Time	The most recent date the data was retrieved from the shared cache.
Dependencies	Lists the assets in the WebCenter Sites database on which the corresponding data depends. If the data is associated with more than one database table, multiple dependencies will be listed for that entry.

## Chapter 10

# Testing Deployed Community Widget Tags

This chapter provides you with instructions for verifying deployed community widget functionality.

This chapter contains the following sections:

- [Sample Site with Community Widgets Deployed](#)
- [Working with Comments and Reviews Widgets as an Authenticated Visitor](#)
- [Working with Comments and Reviews Widgets as an Anonymous Visitor](#)
- [Sorting Comments and Reviews](#)
- [Replying to Posted Comments](#)
- [Viewing Review Details](#)
- [Ranking Visitor Comments and Reviews](#)
- [Sharing Comments and Reviews with Facebook and Twitter](#)
- [Creating and Sharing a Permalink](#)
- [Working with Ratings Widgets](#)
- [Working with Poll Widgets](#)

## Sample Site with Community Widgets Deployed

**Figure 1** is an example of a web page on which a Comments widget tag is deployed. In this example, the Comments widget contains an embedded Login Bar widget. If the Comments widget is configured to allow visitors to post comments to the web page, you will see a **Post a Comment** link which you click to display the comment field and/or the **Login** and **Register** links.

If administrators and moderators have configured the visitor permissions to allow anonymous visitors to post comments to the website, the comment field is displayed when you click the **Post a Comment** link. If the widget's visitor permissions are configured to allow only authenticated visitors to post comments, then visitors must log in to see the comment field.

**Figure 1:** Sample web page with deployed widgets

The screenshot shows a web page for 'force sport'. The main article is 'An Ace is Born with Comments' by John Candy, dated January 17, 2009. The page has a sidebar with various widget categories and a main content area with a photo of a tennis player. A 'Post a comment' link is highlighted with an orange box. Below it, a form is displayed with 'Login | Register' links, a text area for the comment, and 'Preview' and 'Submit' buttons. At the bottom, there is a section for '(1) Comments' showing a guest comment.

**Post a Comment** widget displayed at the bottom of the page.

**Comment** field: Enter the content of your comment in this field.

Click the **Post a Comment** link to display the comments widget.

Approved comments displayed on the web page.

**Post a comment**

[Login](#) | [Register](#)

Your email address will never be displayed or shared with third parties. Your name will be displayed with your comment.

Comment

(1) Comments

[Post a comment](#)



Guest

December 27 Tuesday, 2011 09:40 AM

This is a comment posted by an anonymous visitor

These links are automatically encoded in the primary Login Bar widget tag. If "Login Bar Required" is set to **Yes**, a Login Bar widget is displayed with the Comments widget.

Click the desired action:

- **Login**
- **Register**

## Working with Comments and Reviews Widgets as an Authenticated Visitor

This section provides instructions for logging in to the website on which community widget tags are deployed, and working with the widgets as an authenticated visitor.

### Note

The permissions authenticated visitors have to community widgets are determined by the configurations made to the widget tags in the Community interface. Therefore, certain instructions in this section may not be applicable to you.

This section contains the following topics:

- [Logging in to the Site as an Authenticated Visitor](#)
- [Posting Comments as an Authenticated Visitor](#)
- [Posting Reviews as an Authenticated Visitor](#)
- [Working with Comments as an Authenticated Visitor](#)
- [Working with Reviews as an Authenticated Visitor](#)

### Logging in to the Site as an Authenticated Visitor

Logging in actuates the permissions and settings for authenticated visitors that were configured for the community widget tags in the Community interface.

#### To log in to the site

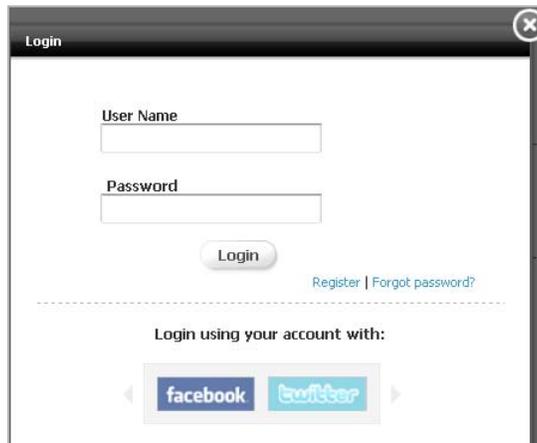
1. Access the web page on which community widget tags are deployed.
2. Log in to the website.
  - If the Login Bar is a standalone widget, navigate to the Login Bar widget and click the **Login** link.
  - If the Login Bar is embedded in the tag of the Comments or Reviews widget, navigate to the Comments or Reviews widget and click either **Post a comment** or **Post a review**. Then click **Login**.

Log in using your assigned credentials. **Log in credentials are case-sensitive.**

You can also use the following default credentials to log in as an authenticated visitor:

**User Name** – demouser

**Password** – demopass

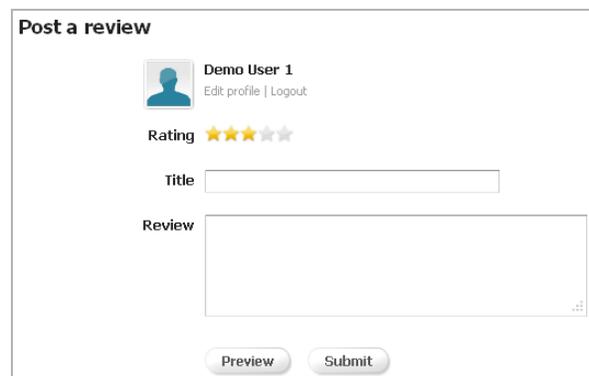
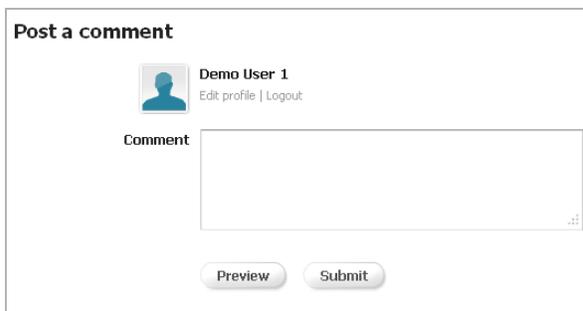


### Note

If the Login Bar widget is configured to allow visitors to log in using credentials for third-party websites, visitors can log in to the website with credentials for social networks such as Facebook.

### 3. Click **Login**.

The “Comment” or “Review” field is displayed:



### 4. Now that you are logged in, you can verify that the community widgets are configured with the appropriate permissions and moderation settings for authenticated visitors.

## Posting Comments as an Authenticated Visitor

To determine whether the commenting permissions and moderation settings configured for authenticated visitors are deployed successfully, log in to the site and post a comment.

### To post a comment as an authenticated visitor

1. Access a web page on which a Comments widget is deployed.
2. Log in to the website with your assigned credentials.

The “Comment” field is displayed:



3. In the “Comment” field, enter your comment.
4. (Optional) Click **Preview** to preview your comment before posting it to the website.



5. Click **Submit**.
  - If pagination is enabled, and the sort order is **Oldest at the top (Comments > Configure > Appearance)**, the comment you post is displayed as the last comment on the first page. This enables you to view the comment you just posted. Refreshing the page will place the comment in the appropriate order.
  - The site’s moderation settings (**Settings > Moderation**) determine whether an authenticated visitor’s comment is automatically approved and displayed on the website, or if all comments must be manually approved before they are displayed on the website. For information about configuring moderation settings, see [“Settings > Moderation,” on page 158](#).
  - If the comment is posted by a blacklisted visitor, then the comment must be manually approved before it is displayed to other visitors on the website. The comment is marked with a status of **Inappropriate** in the Community interface (**Comments > Moderate**). For information about the “Visitor Blacklist,” see [“Settings > Visitor Blacklist,” on page 155](#).
  - If the comment is posted by a whitelisted visitor, then the comment is automatically approved and displayed on the website. For information about the “Visitor Whitelist,” see [“Settings > Visitor Whitelist,” on page 157](#).

## Posting Reviews as an Authenticated Visitor

To determine whether the reviewing permissions and moderation settings configured for authenticated visitors are deployed successfully, log in to the site and post a review.

### To post a review as an authenticated visitor

1. Access the web page on which the Reviews widget is deployed.
2. Log in to the website with your assigned credentials.

The “Review” field is displayed:

**Post a review**

Demo User 1  
Edit profile | Logout

Rating ★★★★★

Title

Review

Preview Submit

Rating field – Click the stars to rate the topic you are reviewing.

Title field – enter a title summarizing the review.

Reviews widget submit box

3. Fill in the fields. The fields you see are determined by the configurations made to the Reviews widget tag before deployment (**Reviews > Configure > Appearance**):
  - **Rating** field – Rate the topic by clicking the desired amount of stars.
  - **Title** field – Enter a title for your review. The title will be displayed above the content of your review.
  - **Review** field – Enter the content of your review in this field.
4. (Optional) Click **Preview** to preview your review before posting it to the website.

Preview

Demo User 1  
December 27 Tuesday, 2011 02:36 PM

★★★★★ New Review

Review posted by an authenticated visitor.

Submit Cancel

5. Click **Submit**.
  - If pagination is enabled, and the sort order is **Oldest at the top (Reviews > Configure > Appearance)**, the review you post is displayed as the last review on the first page. This enables you to view the review you just posted. Refreshing the page will place the review in the appropriate order.
  - The site’s moderation settings determine whether an authenticated visitor’s review is automatically displayed on the website, or if all reviews must be manually

approved before they are displayed on the website. For information about configuring moderation settings, see “[Settings > Moderation](#),” on page 158.

- If the review is posted with a low rating (two stars or less), and the moderation settings are set to mark all reviews given a rating of two stars or less as **Inappropriate**, then the review is marked with a status of **Inappropriate** in the Community interface (**Reviews > Moderate**) and must be manually approved before it is displayed on the website.
- If the review is posted by a blacklisted visitor, then the review must be manually approved before it is displayed on the website. The review is marked with a status of **Inappropriate** in the Community interface (**Reviews > Moderate**). For information about the “Visitor Blacklist,” see “[Settings > Visitor Blacklist](#),” on page 155.
- If the review is posted by a whitelisted visitor, then the review is automatically approved and displayed on the website. For information about the “Visitor Whitelist,” see “[Settings > Visitor Whitelist](#),” on page 157.

## Working with Comments as an Authenticated Visitor

The actions authenticated visitors can perform on comments displayed on the website are determined by the visitor permissions, appearance settings, and moderation settings configured for the Comments widget tag.

This section contains the following:

- [Modifying Comments on the Website](#)
- [Deleting Comments from the Website](#)

### Modifying Comments on the Website

If the Comments widget is configured to enable authenticated visitors to modify the comments they post (**Comments > Configure > Permissions**), then authenticated visitors can modify their own comments directly on the website.

#### To modify a comment on the website

1. Access the web page on which you posted the comment and log in with your assigned credentials.
2. Mouse over the comment you wish to modify. In the menu that appears, click **Edit**.



3. In the “Edit” box, modify the comment.

The screenshot shows a user interface for editing a comment. At the top left, there is a profile picture and the name 'Demo User 1' with the timestamp 'December 27 Tuesday, 2011 02:29 PM'. Below this is a dashed-line box titled 'Edit' containing a text area with the text 'This is a comment posted by an authenticated visitor.' At the bottom of the dialog are three buttons: 'Preview', 'Submit', and 'Cancel'.

4. (Optional) Click **Preview** to preview how the modified comment will look on the website.
5. Click **Submit**.

The site’s moderation settings determine whether the modified comment is automatically approved and displayed on the website, or if it must be manually approved before being displayed. For information about configuring moderation settings, see “[Settings > Moderation](#),” on page 158.

## Deleting Comments from the Website

If the Comments widget tag is configured to enable authenticated visitors to delete the comments they post (**Comments > Configure > Permissions**), then authenticated visitors can delete their own comments directly from the website.

### To delete a comment from the website

1. Access the web page on which you posted the comment and log in with your assigned credentials.
2. Mouse over the comment you wish to delete. In the menu that appears, click **Delete**.



3. Click **Yes** to delete the comment.

The screenshot shows a confirmation dialog box titled 'Delete' with a red 'X' icon. The text inside asks 'Are you sure you want to delete this comment?' At the bottom are two buttons: 'Yes' and 'No'.

The comment is deleted from both the website and the Community interface (**Comments > Moderate**).

## Working with Reviews as an Authenticated Visitor

The actions authenticated visitors can perform on reviews displayed on the website are determined by the visitor permissions, appearance settings, and moderation settings configured for the Reviews widget tag.

This section contains the following topics:

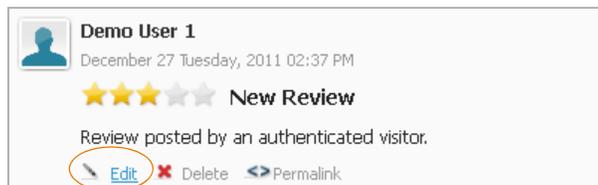
- [Modifying Reviews on the Website](#)
- [Deleting Reviews from the Website](#)

### Modifying Reviews on the Website

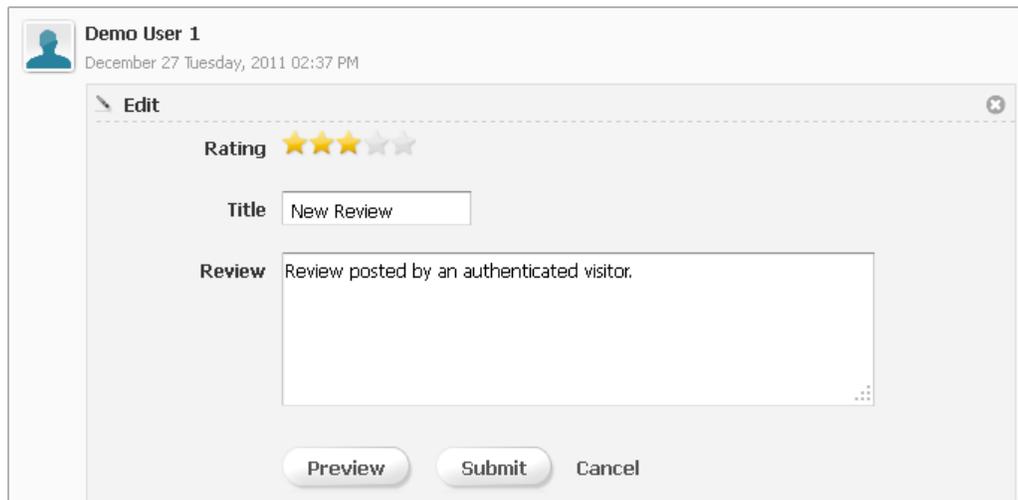
If the reviews widget tag is configured to enable authenticated visitors to modify the reviews they post (**Reviews > Configure > Permissions**), then authenticated visitors can modify their own reviews directly on the website.

#### To modify a review on the website

1. Access the web page on which you posted the review and log in with your assigned credentials.
2. Mouse over the review you wish to modify. In the menu that appears, click **Edit**.



3. Modify the content in the desired fields.



4. (Optional) Click **Preview** to preview how the modified review will look on the website.
5. Click **Submit**.

The site's moderation settings determine whether the modified review is automatically approved and displayed on the website, or if it must be manually approved before

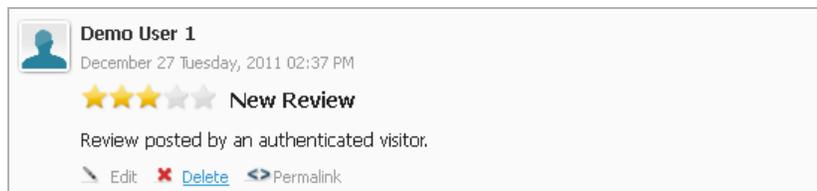
being displayed on the website. For information about configuring moderation settings, see “[Settings > Moderation](#),” on page 158.

## Deleting Reviews from the Website

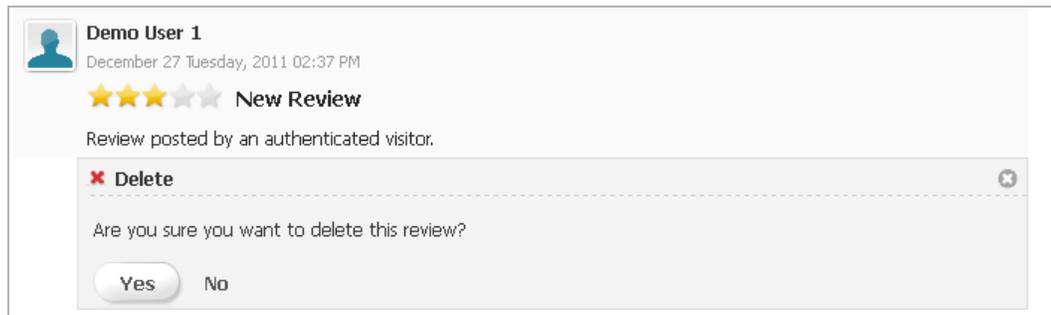
If the Reviews widget tag is configured to enable authenticated visitors to delete their own reviews (**Reviews > Configure > Permissions**), then authenticated visitors can delete their own reviews directly from the website.

### To delete a review from the website

1. Access the web page on which you posted the review you wish to delete, and log in with your assigned credentials.
2. Mouse over the review you wish to delete. In the menu that appears, click **Delete**.



3. Click **Yes** to delete the review.



The review is deleted from both the website and the Community interface (**Reviews > Moderate**).

## Working with Comments and Reviews Widgets as an Anonymous Visitor

This section provides instructions for working with community widgets as an anonymous visitor. The actions you can perform as an anonymous visitor are determined by the configurations made to the community widget tags in the Community interface.

This section contains the following topics:

- [Posting Comments as an Anonymous Visitor](#)
- [Posting Reviews as an Anonymous Visitor](#)

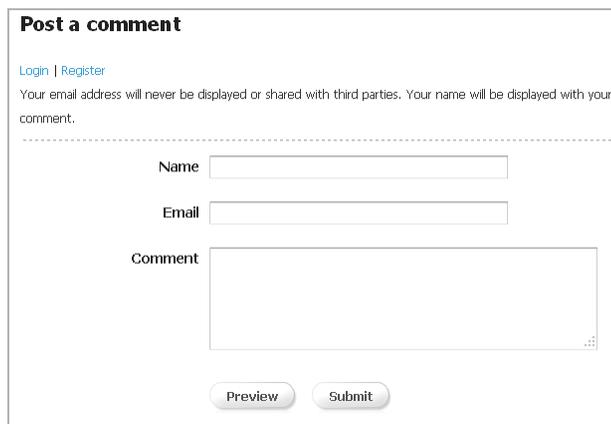
### Posting Comments as an Anonymous Visitor

To ensure that the commenting permissions (**Comments > Configure > Permissions**) for anonymous visitors are configured and deployed successfully, post a comment to the website as an anonymous visitor.

#### To post a comment as an anonymous visitor

1. Access the web page on which the Comments widget tag is deployed.
2. Navigate to the “Comments” section and click the **Post a Comment** link.

If anonymous visitors can post comments, a “Post a Comment” field is displayed:



**Post a comment**

[Login](#) | [Register](#)

Your email address will never be displayed or shared with third parties. Your name will be displayed with your comment.

Name

Email

Comment

3. Fill in the required fields. The fields you see are determined by the configurations made to the Comments widget tag in the Community interface:
  - **Name** and **Email** fields – If a name or email address is required, enter your information into the field(s) provided.
  - **Comment** field – Enter the content of your comment.
4. Click **Submit**.
  - If pagination is enabled, and the sort order is **Oldest at the top (Comments > Configure > Appearance)**, the comment you post is displayed as the last comment on the first page. This enables you to view the comment you just posted. Refreshing the page will place the comment in the appropriate order.
  - The site’s moderation settings determine whether all comments are automatically approved and displayed on the website, or if comments posted by anonymous

visitors must be manually approved before they are displayed on the website. For information about configuring moderation settings, see “[Settings > Moderation](#),” on page 158.

- If the comment is posted by an anonymous visitor whose IP address is blacklisted, then the comment must be manually approved before it is displayed on the website. The comment is marked with a status of **Inappropriate** in the Community interface (**Comments > Moderate**). For information about the “Visitor Blacklist,” see “[Settings > Visitor Blacklist](#),” on page 155.

## Posting Reviews as an Anonymous Visitor

To ensure that the reviewing permissions (**Reviews > Configure > Permissions**) for anonymous visitors are configured and deployed successfully, post a review to the website as an anonymous visitor.

### To post a review as an anonymous visitor

1. Access a web page on which the Reviews widget tag is deployed.
2. Navigate to the “Reviews” section of the page and click the **Post a Review** link.

**Post a review**

[Login](#) | [Register](#)

Your email address will never be displayed or shared. Your name will be displayed with your review.

---

Name

Email

Rating ★★★★★

Title

Review

3. Fill in the required fields. The fields you see are determined by the configurations made to the Reviews widget tag in the Community interface:
  - **Name** and **Email** fields – If a name and/or email address is required, enter your information into the field(s) provided.
  - **Rating** field – Rate the topic you are reviewing by clicking the desired amount of stars.
  - **Title** field – Enter a title summarizing the content of your review.
  - **Review** field – Enter the content of your review.
4. Click **Submit**.
  - If pagination is enabled, and the sort order is **Oldest at the top** (**Reviews > Configure > Appearance**), the review you post is displayed as the last review on the first page. This enables you to view the review you just posted. Refreshing the page will place the review in the appropriate order.

- The site's moderation settings determine whether all reviews are automatically approved and displayed on the website, or if reviews posted by anonymous visitors must be manually approved before being displayed on the website. For more information about configuring moderation settings, see "[Settings > Moderation](#)," on page 158.
- If the review is posted with a low rating (two stars or less), and the moderation settings are set to mark all reviews given a rating of two stars or less as **Inappropriate**, then the review is marked with a status of **Inappropriate** in the Community interface (**Reviews > Moderate**) and must be manually approved before it is displayed on the website.
- If the review is posted by an anonymous visitor whose IP address is blacklisted, then the review must be manually approved before it is displayed on the website. The review will be marked with a status of **Inappropriate** in the Community interface (**Reviews > Moderate**). For information about the "Visitor Blacklist," see "[Settings > Visitor Blacklist](#)," on page 155.

## Sorting Comments and Reviews

When browsing comments (or reviews) on a web page, you can choose the order in which the comments (or reviews) are displayed.

### To sort comments (or reviews)

1. Access a web page on which the Comments (or Reviews) widget tag is deployed.
2. Navigate to the Comments (or Reviews) widget. Click the **Sort by** drop-down menu, and select one of the following options:
  - **Newest at the top** – Displays the most recently posted comments or reviews at the top of the list.
  - **Oldest at the top** – Displays the least recent comments or reviews at the top of the list.
  - **Most ranked at the top** – Displays the comments or reviews that site visitors rate frequently at the top of the list.
  - **Least ranked at the top** – Displays the comments or reviews that site visitors rate the least at the top of the list.

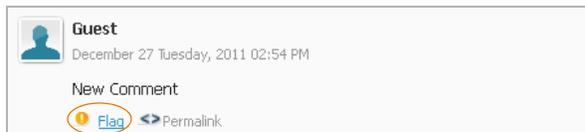
The comments (or reviews) are displayed in the order you specify.

## Flagging Inappropriate Posts

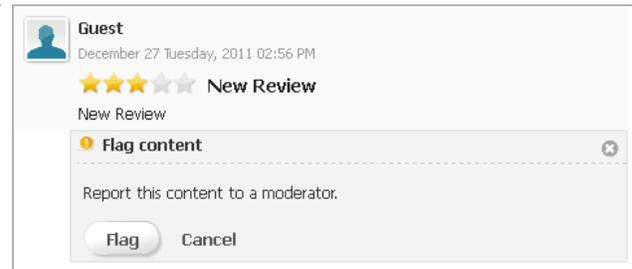
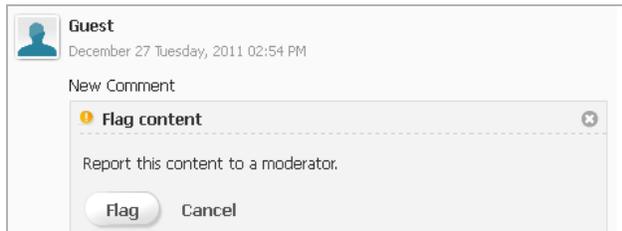
If a comment or review contains inappropriate or inaccurate content, visitors can assist in the moderation process by flagging that particular post. The type of visitors (authenticated and anonymous) who can flag comments and reviews is determined by the flagging permissions configured in the Community interface (**Settings > Moderation**).

### To flag a comment or review on the website

1. Access the web page on which the comment (or review) you wish to flag is displayed.
2. If you are an authenticated visitor, log in using your assigned credentials. If you are an anonymous visitor, skip to [step 3](#).
3. Mouse over the comment (or review) that contains the inappropriate or inaccurate content, and click **Flag**.



4. In the “Flag Content” box, click **Flag**.



You have successfully flagged the comment (or review). Once an administrator or moderator accesses the Community interface (**Comments > Moderate** or **Reviews > Moderate**), he will see **Flagged** next to the approved post's status, and can then take the appropriate actions to either remove or edit the post.

## Replying to Posted Comments

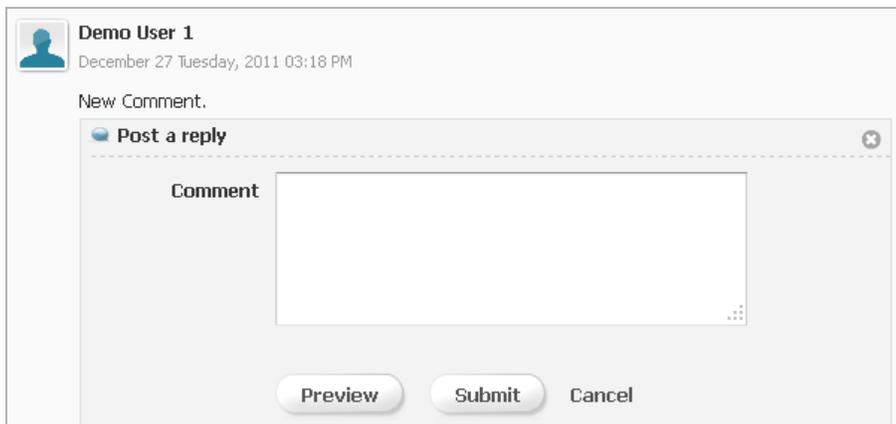
If the comments widget tag is configured to support a threaded commenting structure (**Comments > Configure > Appearance**), then visitors can post replies to comments that are displayed on the website. The type of visitors (authenticated and anonymous) who can reply to comments is determined by the visitor permissions configured in the Community interface (**Comments > Configure > Permissions**).

### To reply to a comment

1. Access the web page that contains the comment to which you wish to post a reply.
2. If you are an authenticated visitor, log in with your assigned credentials. If you are an anonymous visitor, skip to [step 3](#).
3. Mouse over the comment to which you wish to post a reply. In the menu that appears, click **Reply**.



4. In the “Post a Reply” section, type your reply to the comment. If you are posting a reply as an anonymous visitor, you may have to enter your name, email address, and/or a verification code depending on the configurations made to the Comments widget tag in the Community interface.



5. (Optional) Click **Preview** to preview how the response will look on the website.
6. Click **Submit**.



Expand to view the replies posted to the comment. Collapse to hide the replies.

Your post is displayed under the comment to which you replied. If this is a moderated discussion, then the reply must be approved before it is displayed on the website.

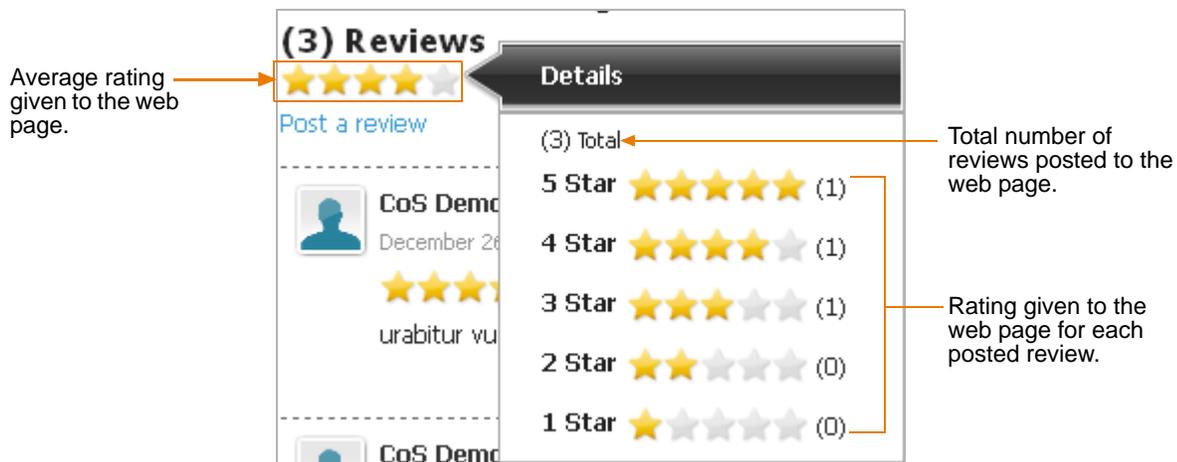
## Viewing Review Details

If the Reviews widget is configured to allow visitors to post a rating with their review, the Reviews widget displays a rating field. The average rating (in the form of stars) given to the web page (topic) on which this Reviews widget is deployed, is displayed under the amount of reviews posted to the web page. Point to the stars to view the “Details” pop-up which displays the total number of reviews posted to the web page along and a summary of the rating each visitor submitted with his review.

### To view review details

1. Access a web page on which the Reviews widget is deployed.
2. Navigate to the “Reviews” section of the page and mouse over the stars (average rating given to the topic) displayed next to it.

The “Details” pop-up box is displayed.



## Ranking Visitor Comments and Reviews

If the Comments and Reviews widgets are configured to support individual rankings by visitors, then visitors can rank the content of an individual comment or review by indicating whether the comment or review is helpful.

### To rate individual comments or reviews

1. Access the web page to which the comment or review you wish to rank is posted.
2. If you are an authenticated visitor, log in with your assigned credentials. If you are an anonymous visitor, skip to [step 3](#).
3. Navigate to the comment or review you wish to rank. In the “Was this helpful?” field, select either **Yes** or **No**.



## Sharing Comments and Reviews with Facebook and Twitter

If the Login Bar widget is configured to allow visitors to share comments and reviews via Facebook and Twitter (**Login Bar > Configure**), and a visitor has an account with either of these social network providers, then that visitor can share a link on Facebook or Twitter to the web page on which the comment or review is posted.

### To share a comment or review via Facebook or Twitter

1. Access the web page to which the comment or review you wish to share is posted.
2. Point to the comment or review you wish to share via Facebook or Twitter. In the menu that appears, click one of the icons in the “Share to” option. Click either the **Facebook** (  ) or **Twitter** (  ) icon.

For example:



3. In the pop-up window, enter your credentials for the social network provider with which you chose to share the comment or review.

A link to the web page is posted to your Facebook or Twitter news feed. Clicking this link opens the web page to which the comment or review is posted.

## Creating and Sharing a Permalink

A permalink is a URL that is used to refer to a specific comment or review on a web page. When you access this URL, a static version of the web page opens displaying the desired comment or review as the main focus of the page. Visitors can view and copy the link for a particular comment or review by clicking the **Permalink** option, located below the desired comment or review.

### To use the permalink option

1. Access the web page to which the desired comment or review is posted.  
If the widget requires you to log in to work with the widget, see [step 2](#). If any visitor can work with the widget, skip to [step 3](#).
2. Navigate to the login bar and click Login. Log in with your assigned credentials.
3. Point to the desired comment or review.
4. In the menu that appears, click **Permalink**.



The address bar displays the URL of the web page with the appended ID of the desired comment or review.

5. Copy the URL displayed in the address bar.  
You can share this URL by posting it to a news feed, sending it to a contact through email, and so on.

## Working with Ratings Widgets

If the widgets related to rating functionality are configured to enable authenticated visitors to rate topics on the website (**Ratings > Configure > Permissions**), only authenticated visitors can use the ratings widgets deployed on the pages of the website. Otherwise, all visitors (authenticated and anonymous) can use the ratings widgets. To determine whether the visitor permissions for rating content on the website are deployed successfully, follow the instructions in this section.

### To submit a rating for a page asset

1. Access the web page on which a ratings widget is deployed. The type of ratings widgets you may see are: Stars Ratings, Thumbs Up/Down Ratings, Like It Ratings, and Recommend Ratings.

If the ratings widget requires visitors to log in to the website in order to submit a rating, see [step 2](#). If all visitors can rate the content of an asset, skip to [step 3](#).

2. Navigate to the login bar and click **Login**. Log in with your assigned credentials.
3. Submit a rating for the topic. The type of rating field that is displayed to you is determined by which ratings widget tag was deployed on the web page. The following is a list of rating fields you may see on a web page:
  - **Stars Ratings** – (Complex rating) This widget displays a stars rating field which you can use to rate the content of a topic by selecting from a range of one to five stars. One star represents the lowest rating that can be given to a topic and five stars represents the highest rating.

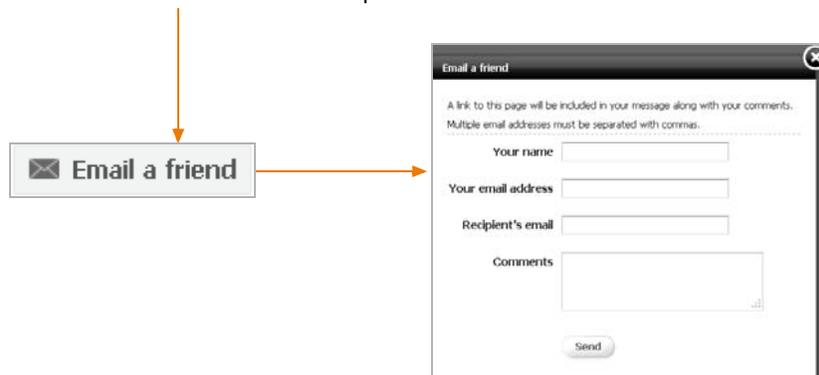


- **Thumbs Up/Down Ratings** – (Binary rating) This widget displays thumbs up and thumbs down icons. Click one of the icons to indicate whether the content of the topic is informative.



- **Recommend Ratings** – (Unary rating) This widget displays an **Email a Friend** link which you can click to email the URL of the web page to a contact who may find the content useful.

Click the **Email a Friend** link to open the “Email to a Friend” form.



- **Like It Ratings** – (Unary rating) This widget displays a **Like** link which you can click to indicate that you liked the content of a given topic.

When you click the **Like** link, the link becomes unclickable and indicates that you like the content.

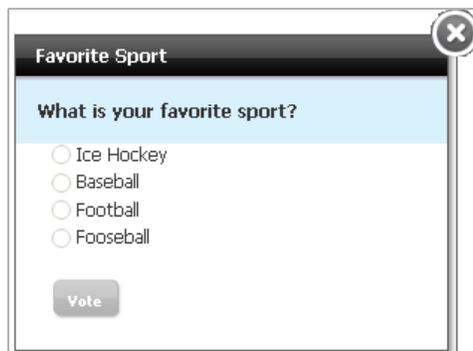


## Working with Poll Widgets

Poll widgets are used to survey site visitors on a particular topic. Each visitor can submit one vote per poll.

### To work with a Poll widget

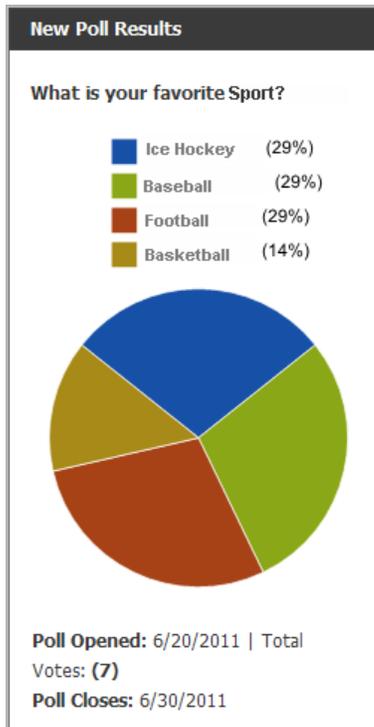
1. Access the web page on which a Poll Tag is deployed.
2. Navigate to the poll widget and select an answer for the poll's question.



3. Click **Vote**.
  - If “Show Results” is disabled (“[Creating a New Poll](#),” on page 146), then the poll question and its answer options are re-displayed to site visitors. However, the answer options are greyed out, ensuring that each visitor submits only one vote.
  - If “Show Results” is enabled and the “Results View” field is set to **PopUp** (“[Creating a New Poll](#),” on page 146), the **View Poll Results** button is displayed inside the Poll widget. Click the **View Poll Results** button to view a summary of the poll's results in a pop up box.
  - If show results is enabled and the “Results View” field is set to **In-Site** (“[Creating a New Poll](#),” on page 146), a summary of the poll's results is displayed inside the Poll widget.

### To work with the poll Results widget

Access the web page on which the poll's Results Tag is deployed and navigate to the poll Results widget. Below is an example of a poll results widget configured to display the summary of the poll's results as a pie chart.





## Appendix A

# Analyzing Community Widget Tags

Each widget has its own deployment screen which displays the widget's tag. This appendix provides information about the parameters defined in each widget tag.

This appendix contains the following sections:

- [Comments Widget Tags](#)
- [Reviews Widget Tags](#)
- [Ratings Widget Tags](#)
- [Login Bar Widget Tag](#)
- [Poll Widget Tags](#)

## Comments Widget Tags

The widget tags related to adding commenting functionality to your website are the following:

- [Comments Widget Tag](#)
- [Comments Summary Widget Tag](#)
- [Links to Topics Widget Tag](#)
- [Recently Commented Widget Tag](#)
- [Most Commented Widget Tag](#)

### Comments Widget Tag

The “Comments Deployment” screen (**Comments > Deploy > Comments**) provides administrators and designers with the Comments widget tag. This section analyzes the parameters defined in the Comments widget tag:

#### Comments widget tag

```

1     <div id="comments_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.comments",
6             version: "1.5",
7             elementID: "comments_container",
8             attributes: {"site_id":"FirstSiteII"}});
9         setTimeout(
10            function(){if ((typeof(wsdk) == 'undefined')
11                || (typeof(wsdk.comments) == 'undefined'))
12                {document.getElementById('comments_container').innerHTML
13                = "<div style='font-family: Tahoma, Verdana, Geneva,
14                sans-serif;font-size: 12px;color: #333333;border: 1px
15                solid #dbdfef;padding-left: 5px;padding-top: 4px;height:
16                22px;'>Comments is unavailable right now. Please contact
17                the site administrator.</div>";}}
18            ,30000);

19        cos.pageScripts = cos.pageScripts || [];
20        cos.pageScripts.push('wsdk.comments');

21        (function()
22            {
23                var oldOnloadHandler = window.onload || function()
24                {
25                };
26                if (!oldOnloadHandler.alreadyProcessed)
27                {
28                    window.onload = function()
29                    {

```

```

23         var script =
                document.createElement('script');
24         script.src = 'http://
                cosservice.fatwire.com:8080/cos/wsdk/
                widget/'
25                 + cos.pageScripts.join(':') +
                '.js?site_id=FirstSiteII';
26         script.type = 'text/javascript';
27         script.charset = 'utf-8';
28         document.getElementsByTagName("head").item(0)
                .appendChild(script);
29         oldOnloadHandler.apply(this, arguments);
30     };
31     window.onload.alreadyProcessed = true;
32 }

33     })();
34 </script>

```

### Analyzing the Comments widget tag

- [Line 1](#) defines the container that holds the Comments widget on the page. If you assign a resource ID to this instance of the Comments widget, the widget's resource ID is also specified in this line. For example:

```

<div id= "comments_container_CommentsResourceID_other">
  </div>

```

Where *CommentsResourceID* is the resource ID assigned to this instance of the Comments widget.

**Note:** If you do not specify a resource ID for this widget tag, the encrypted URL of the page on which this widget tag is deployed is assigned as the widget's resource ID.

- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Comments widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Comments widget inside the container defined in [line 1](#).
- [Lines 9 – 11](#) check if the Comments widget is rendered on the page. If the Comments widget is not rendered after 30 seconds, then users are informed that there is an error and should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Comments widget tag (`wsdk.comments`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is any logic deployed on the page, this line defines the

function that implements it. If there is no such logic deployed, then the function is left empty.

- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 2:** Comments widget displayed on a web page

The screenshot shows a web page for 'force sport'. The main article is titled 'An Ace is Born with Comments' by John Candy. Below the article, there is a 'Comments' widget. The widget includes a search bar, a list of widget pages (Comments All, Comments with ratings, Reviews All, Polls All, Ratings All), a photo of a tennis player, and a 'Post a comment' form. The form has a 'Name' field and a 'Comment' field. Below the form, it says '(0) Comments' and 'Post a comment'. There are also 'Preview' and 'Submit' buttons.

Comments widget displaying a "Name" and "Comment" field.

## Comments Summary Widget Tag

The “Comments Summary Deployment” screen (**Comments > Deploy > Comments Summary**) provides administrators and developers with the Comments Summary widget tag. This section analyzes the parameters defined in the Comments Summary widget tag:

### Comments Summary widget tag

```

1     <div id="comments_summary_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "comments-summary",
6             version: "0.1",
7             elementID: "comments_summary_container",
8             attributes:
9             {"site_id":"FirstSiteII","show_last_comment_date":"false"
10            }});
11            setTimeout(
12                function(){if ((typeof(cos) ==
13                'undefined')||(typeof(comments_summary) == 'undefined'))
14                {document.getElementById('comments_summary_container').in
15                nerHTML = "<div style='font-family: Tahoma, Verdana,
16                Geneva, sans-serif;font-size: 12px;color: #333333;border:
17                1px solid #dbdfef;padding-left: 5px;padding-top:
18                4px;height: 22px;'>Comments Summary is unavailable right
19                now. Please contact the site administrator.</div>";}}
20                ,30000);

21            cos.pageScripts = cos.pageScripts || [];
22            cos.pageScripts.push('comments-summary');

23            (function()
24            {
25                var oldOnloadHandler = window.onload || function()
26                {
27                };
28                if (!oldOnloadHandler.alreadyProcessed)
29                {
30                    window.onload = function()
31                    {
32                        var script = document.createElement
33                        ('script');
34                        script.src = 'http://cosservice.fatwire
35                        .com:8080/cos/wsdk/widget/'
36                        + cos.pageScripts.join(':') +
37                        '.js?site_id=FirstSiteII';
38                        script.type = 'text/javascript';
39                        script.charset = 'utf-8';
40                        document.getElementsByTagName("head")
41                        .item(0).appendChild(script);
42                        oldOnloadHandler.apply(this, arguments);

```

```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Comments Summary widget tag

- [Line 1](#) defines the container that holds the Comments Summary widget on the page. If you specify the resource ID of a Comments widget in this instance of the Comments Summary widget tag, the resource ID for that Comments widget is also specified in this line. For example:

```
<div id= "comments_summary_container_CommentsWidgetResourceID
    _other"></div>
```

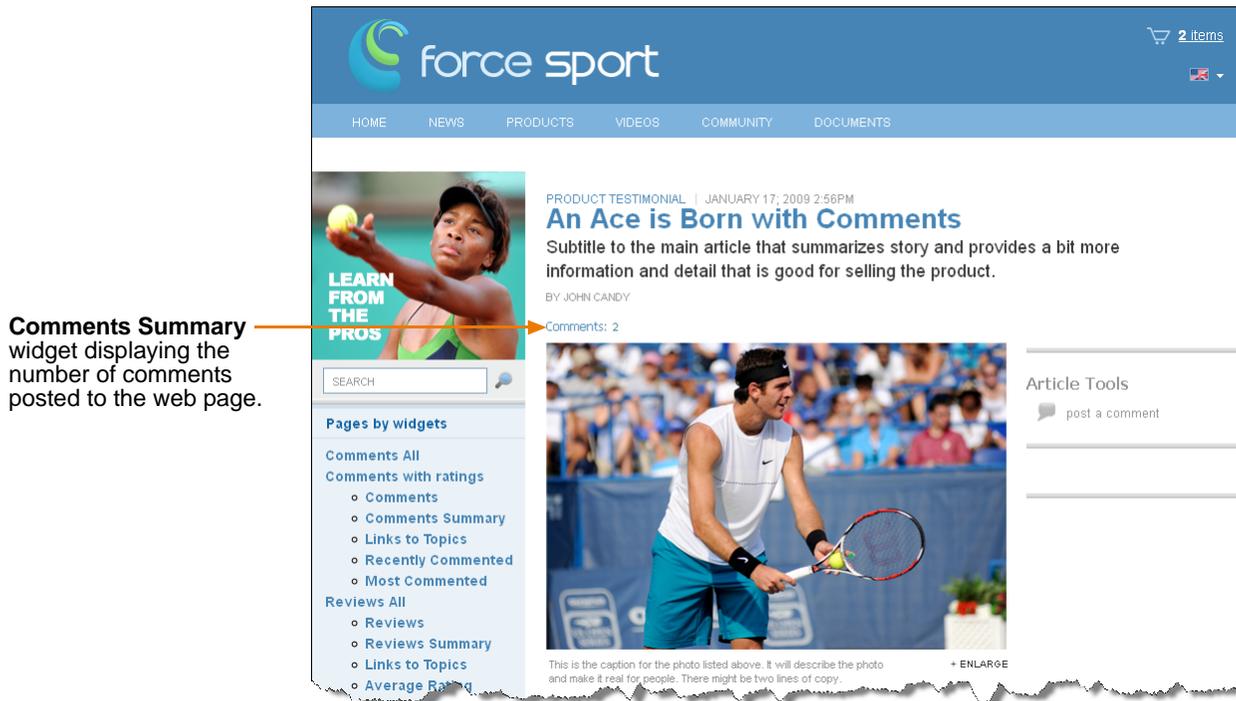
Where *CommentsWidgetResourceID* is the resource ID of the desired Comments widget.

**Note:** If you deploy the Comments Summary widget on a web page associated with a Comments widget, and that Comments widget does not have a resource ID, the encrypted URL of the page is specified in the Comments Summary widget tag as the Comments widget's resource ID.

- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Comments Summary widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Comments Summary widget inside the container defined in [line 1](#).
- [Lines 9 – 11](#) check if the Comments Summary widget is rendered on the page. If the Comments Summary widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Comments Summary widget tag (`comments-summary`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is any logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.

- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 3:** Comments Summary widget displayed on a web page



## Links to Topics Widget Tag

The “Links to Topics Deployment” screen (**Comments > Deploy > Links to Topics**) provides administrators and developers with the Links to Topics widget tag. This section analyzes the parameters defined in the Links to Topics widget tag:

### Links to Topics widget tag

```

1      <a href="http://host:port/test/page1.html#comments_link
      &resource_id=CommentsID"></a>
2      <div id="comments_link_div"></div>
3      <script type="text/javascript">
4          cos = window.cos || {};
5          cos.pageWidgets = cos.pageWidgets || [];

6          cos.pageWidgets.push({name: "comments-link",
7              version: "0.1",
8              elementID: "comments_link_div",
9              attributes: {"site_id":"FirstSiteII"}});

10         cos.pageScripts = cos.pageScripts || [];
11         cos.pageScripts.push('comments-link');

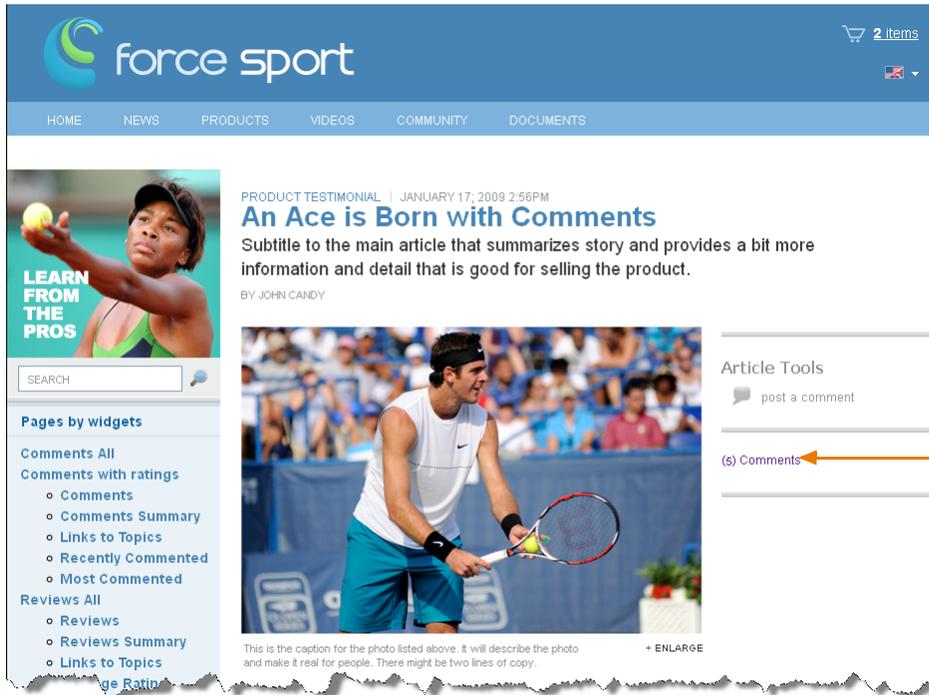
12         (function()
13         {
14             var oldOnloadHandler = window.onload || function()
15             {
16             };
17             if (!oldOnloadHandler.alreadyProcessed)
18             {
19                 window.onload = function()
20                 {
21                     var script = document.createElement
22                         ('script');
23                     script.src = 'http://cosservice.fatwire
24                         .com:8080/cos/wsdk/widget/'
25                         + cos.pageScripts.join(':') +
26                         '.js?site_id=FirstSiteII';
27                     script.type = 'text/javascript';
28                     script.charset = 'utf-8';
29                     document.getElementsByTagName("head")
30                         .item(0).appendChild(script);
31                     oldOnloadHandler.apply(this, arguments);
32                 };
33                 window.onload.alreadyProcessed = true;
34             }
35         })();
36     </script>

```

**Analyzing the Links to Topics widget tag**

- **Line 1** is a link to the web page on which the desired Comments widget tag is deployed, and must be added when the Links to Topics tag is inserted into the desired template.
- **Line 2** defines the container that holds the Links to Topics widget on the page.
- **Line 3** opens the bootstrapping JavaScript code needed for the widget.
- **Line 4** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 5** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 6 – 9** define the Links to Topics widget tag and add the new element to the list of widget tags (defined in **line 5**). **Line 8** contains a link that is used to render the Links to Topics widget inside the container defined in **line 2**.
- **Line 10** defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is left empty.
- **Line 11** adds the Links to Topics widget tag (`comments-link`) to the list of page scripts (defined in **line 10**).
- **Line 14** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- **Line 17** checks if the `onload` function is already processed.
- **Line 19** overrides the code that runs when the page is loaded.
- **Lines 21 – 26** add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- **Line 27** adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- **Line 29** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- **Line 32** closes the bootstrapping JavaScript code needed for the widget.

Figure 4: Links to Topics widget displayed on a web page



The **Links to Topics** widget displaying the number of comments posted to a given web page. Click this link to view the comments on the web page.

## Recently Commented Widget Tag

The “Recently Commented Deployment” screen (**Comments > Deploy > Recently Commented**) provides administrators and designers with the Recently Commented widget tag. This section analyzes the parameters defined in the Recently Commented widget tag:

### Recently Commented widget tag

```

1     <div id="topics_container_recently_commented_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.topics",
6         version: "1.5",
7         elementID: "_recently_commented_container",
8         attributes: {"content_type":"recently_commented",
9         "resource_type":"other","site_id":"FirstSiteII","count_topi
10        cs":"10"}});
11        setTimeout(
12            function(){if ((typeof(wsdk) == 'undefined') ||
13            (typeof(wsdk.topics) == 'undefined'))
14            {document.getElementById('topics_container_recently_comment
15            ed_other').innerHTML = "<div style='font-family: Tahoma,
16            Verdana, Geneva, sans-serif;font-size: 12px;color:
17            #333333;border: 1px solid #dbdfef;padding-left:
18            5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19            right now. Please contact the site administrator.</div>";}}
20            ,30000);

21        cos.pageScripts = cos.pageScripts || [];
22        cos.pageScripts.push('wsdk.topics');

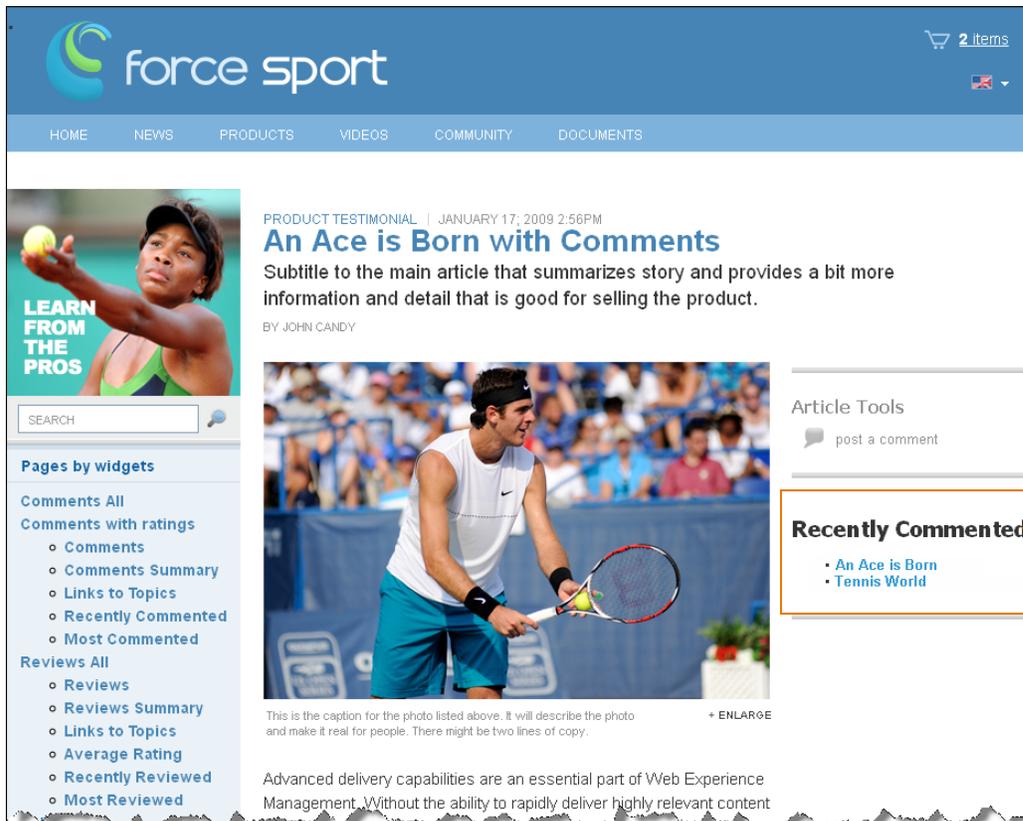
23        (function()
24        {
25            var oldOnloadHandler = window.onload || function()
26            {
27            };
28            if (!oldOnloadHandler.alreadyProcessed)
29            {
30                window.onload = function()
31                {
32                    var script = document.createElement
33                    ('script');
34                    script.src = 'http://cosprod.ua.oracle.com/
35                    cos/wsdk/widget/'
36                    + cos.pageScripts.join(':') +
37                    '.js?site_id=FirstSiteII';
38                    script.type = 'text/javascript';
39                    script.charset = 'utf-8';
40                    document.getElementsByTagName("head")
41                    .item(0).appendChild(script);

```

```
29             oldOnloadHandler.apply(this, arguments);
30         };
31         window.onload.alreadyProcessed = true;
32     }
33     })();
34 </script>
```

### Analyzing the Recently Commented widget tag

- [Line 1](#) defines the container that holds the Recently Commented widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Recently Commented widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Recently Commented widget inside the container defined in [line 1](#). [Line 8](#) specifies the Recently Commented widget tag's settings, including the type of topics the Recently Commented widget will list. If you want the Recently Commented widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Recently Commented widget is rendered on the page. If the Recently Commented widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Recently Commented widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, which includes all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 5:** Recently Commented widget displayed on a web page

Title for this widget

Recently Commented widget displaying a list of web pages to which visitors have recently posted comments.

**Note**

When the Recently Commented widget is deployed, it does not display a title (as shown in [Figure 5](#)). For the Recently Commented widget to display a title, you will have to insert the title into the template to which you copied the Recently Commented widget tag.

## Most Commented Widget Tag

The “Most Commented Deployment” screen (**Comments > Deploy > Most Commented**) provides administrators and designers with the Most Commented widget tag. This section analyzes the parameters defined in the Most Commented widget tag:

### Most Commented widget tag

```

1     <div id="topics_container_most_commented_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5     cos.pageWidgets.push({name: "wsdk.topics",
6     version: "1.5",
7     elementID: "_most_commented_other_container",
8     attributes: {"content_type":"most_commented",
9         "resource_type":"other" "site_id":"FirstSiteII", "count_top
10        ics":"10"}});
11     setTimeout(
12         function(){if ((typeof(wsdk) == 'undefined') ||
13         (typeof(wsdk.topics) == 'undefined'))
14         {document.getElementById('topics_container_most_commented
15         _other').innerHTML = "<div style='font-family: Tahoma,
16         Verdana, Geneva, sans-serif;font-size: 12px;color:
17         #333333;border: 1px solid #dbdfef;padding-left:
18         5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19         right now. Please contact the site administrator.</div>";}}
20         ,30000);

21     cos.pageScripts = cos.pageScripts || [];
22     cos.pageScripts.push('wsdk.topics');

23     (function()
24     {
25         var oldOnloadHandler = window.onload || function()
26         {
27             };
28         if (!oldOnloadHandler.alreadyProcessed)
29         {
30             window.onload = function()
31             {
32                 var script = document.createElement
33                 ('script');
34                 script.src = 'http://cosservice.fatwire.
35                 com:8080/cos/wsdk/widget/'
36                 + cos.pageScripts.join(':') +
37                 '.js?site_id=FirstSiteII';
38                 script.type = 'text/javascript';
39                 script.charset = 'utf-8';
40                 document.getElementsByTagName("head")
41                 .item(0).appendChild(script);
42                 oldOnloadHandler.apply(this, arguments);

```

```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Most Commented widget tag

- [Line 1](#) defines the container that holds the Most Commented widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Most Commented widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Most Commented widget inside the container defined in [line 1](#). [Line 8](#) specifies the Most Commented widget tag's settings, including the type of topics the Most Commented widget will list. If you want the Most Commented widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Most Commented widget is rendered on the page. If the Most Commented widget is not rendered after 30 seconds, users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of page scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Most Commented widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 6:** Most Commented widget displayed on a web page

The screenshot shows a Force Sport website with a main article titled "An Ace is Born with Comments" by John Candy. A "Most Commented" widget is located on the right side of the article. The widget has a title "Most Commented" and lists two items: "An Ace is Born" and "Tennis World". Annotations with arrows point to the widget title and the list of items.

**Title for this widget**

**Most Commented** widget listing the pages to which site visitors frequently post comments.

### Note

When the Most Commented widget is deployed, it does not display a title (as shown in [Figure 6](#)). For the Most Commented widget to display a title, you will have to insert the title into the template to which you copied the Most Commented widget tag.

## Reviews Widget Tags

The widget tags related to adding reviewing functionality to your website are the following:

- [Reviews Widget Tag](#)
- [Reviews Summary Widget Tag](#)
- [Links to Topics Widget Tag](#)
- [Reviews Average Rating Widget Tag](#)
- [Top Ranked Reviews Widget Tag](#)
- [Recently Reviewed Widget Tag](#)
- [Most Reviewed Widget Tag](#)

### Reviews Widget Tag

The “Reviews Deployment” screen (**Reviews > Deploy > Reviews**) provides administrators and developers with the Reviews widget tag. This section analyzes the parameters defined in the Reviews widget tag:

#### Reviews widget tag

```

1     <div id="reviews_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5     cos.pageWidgets.push({name: "wsdk.reviews",
6     version: "1.5",
7     elementID: "reviews_container",
8     attributes: {"site_id":"FirstSiteII"}});
9     setTimeout(
10        function(){if ((typeof(wsdk) == 'undefined') ||
11        (typeof(wsdk.reviews) == 'undefined'))
12        {document.getElementById('reviews_container').innerHTML =
13        "<div style='font-family: Tahoma, Verdana, Geneva, sans-
14        serif;font-size: 12px;color: #333333;border: 1px solid
15        #dbdfef;padding-left: 5px;padding-top: 4px;height:
16        22px;'>Reviews is unavailable right now. Please contact the
17        site administrator.</div>";}}
18        ,30000);

19        cos.pageScripts = cos.pageScripts || [];
20        cos.pageScripts.push('wsdk.reviews');

21        (function()
22        {
23            var oldOnloadHandler = window.onload || function()
24            {
25            };
26            if (!oldOnloadHandler.alreadyProcessed)
27            {

```

```

21         window.onload = function()
22         {
23             var script = document.createElement
24                 ('script');
25             script.src = 'http://coservice.fatwire
26                 .com:8080/cos/wsdk/widget/'
27                 + cos.pageScripts.join(':') +
28                 '.js?site_id=FirstSiteII';
29             script.type = 'text/javascript';
30             script.charset = 'utf-8';
31             document.getElementsByTagName("head")
32                 .item(0).appendChild(script);
33             oldOnloadHandler.apply(this, arguments);
34         };
35     window.onload.alreadyProcessed = true;
36 }
37
38     }}();
39 </script>

```

### Analyzing the Reviews widget tag

- **Line 1** defines the container that holds the Reviews widget on the page. If you assign a resource ID to this instance of the Reviews widget tag, the widget's resource ID is also specified in this line. For example:

```
<div id= "reviews_container_ResourceID_other"></div>
```

Where *ResourceID* is the resource ID assigned to this instance of the Reviews widget.

**Note:** If you do not specify a resource ID for this widget, the encrypted URL of the page on which this widget tag is deployed is assigned as the widget's resource ID.

- **Line 2** opens the bootstrapping JavaScript code needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Reviews widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Reviews widget inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Reviews widget is rendered on the page. If the Reviews widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrators.
- **Line 12** defines the list of page scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Reviews widget tag (`wsdk.reviews`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.

- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag to the page, which includes all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 7:** Reviews widget displayed on a web page

The screenshot displays a web page for 'force sport'. The main content area features an article titled 'An Ace is Born with Comments' with a subtitle 'Subtitle to the main article that summarizes story and provides a bit more information and detail that is good for selling the product.' The article is by John Casey and includes a photo of a tennis player. To the right of the article is an 'Article Tools' section with a 'post a comment' button. Below the article is a 'Post a review' widget. The widget shows the user 'Demo User' with a rating of 5 stars. The form includes a 'Title' field, a 'Review' text area, and 'Preview' and 'Submit' buttons. Below the form, it indicates '(0) Reviews' and provides a 'Post a review' link. The footer of the page contains navigation links and a copyright notice: 'Copyright 2009 2009 by The Company Inc. All rights reserved.'

Reviews widget displaying a "Rating," "Title," and "Review" field.

## Reviews Summary Widget Tag

The “Reviews Summary Deployment” screen (**Reviews > Deploy > Reviews Summary**) provides administrators and designers with the Reviews Summary widget tag. This section analyzes the parameters defined in the Reviews Summary widget tag:

### Reviews Summary widget tag

```

1     <div id="reviews_summary_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "reviews-summary",
6             version: "0.1",
7             elementID: "reviews_summary_container",
8             attributes: {"show_last_comment_date":"false",
9                 "site_id":"FirstSiteII"}});
10        setTimeout(
11            function(){if ((typeof(cos) ==
12                'undefined') || (typeof(cos.reviews-summary) == 'undefined'))
13                {document.getElementById('reviews_summary_container').innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
14                sans-serif;font-size: 12px;color: #333333;border: 1px solid
15                #dbdfef;padding-left: 5px;padding-top: 4px;height:
16                22px;'>Reviews Summary is unavailable right now. Please
17                contact the site administrator.</div>";}}
18            ,30000);

19        cos.pageScripts = cos.pageScripts || [];
20        cos.pageScripts.push('reviews-summary');

21        (function()
22        {
23            var oldOnloadHandler = window.onload || function()
24            {
25            };
26            if (!oldOnloadHandler.alreadyProcessed)
27            {
28                window.onload = function()
29                {
30                    var script = document.createElement
31                        ('script');
32                    script.src = 'http://coservice.fatwire.com
33                        :8080/cos/wsdk/widget/'
34                        + cos.pageScripts.join(':') +
35                        '.js?site_id=FirstSiteII';
36                    script.type = 'text/javascript';
37                    script.charset = 'utf-8';
38                    document.getElementsByTagName("head")
39                        .item(0).appendChild(script);
40                    oldOnloadHandler.apply(this, arguments);
41                }
42            }
43        });

```

```
31         window.onload.alreadyProcessed = true;
32     }
33     })();
34 </script>
```

### Analyzing the Reviews Summary widget tag

- **Line 1** defines the container that holds the Reviews Summary widget on the page. If you specify the resource ID of a Reviews widget in this instance of the Reviews Summary widget tag, the resource ID for that Reviews widget is also specified in this line. For example:

```
<div id= "reviews_summary_container_ReviewsWidgetResourceID
    _other"></div>
```

Where *ReviewsWidgetResourceID* is the resource ID of the desired Reviews widget.

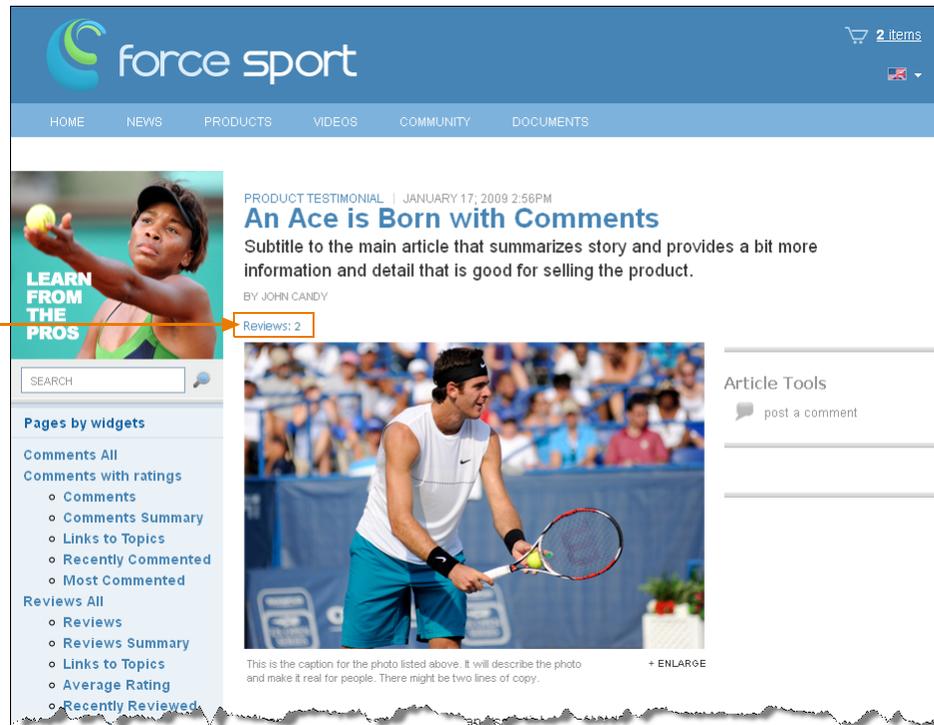
**Note:** If you deploy the Reviews Summary widget on a web page associated with a Comments widget, and that Comments widget does not have a resource ID, the encrypted URL of the page is specified in the Comments Summary widget tag as the Comments widget's resource ID.

- **Line 2** opens the bootstrapping JavaScript code needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Reviews Summary widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Reviews Summary widget inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Reviews Summary widget is rendered on the page. If the Reviews Summary widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- **Line 12** defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Reviews Summary widget tag (`reviews-summary`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no logic deployed, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add the JavaScript tag to the page, which includes all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.

- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 8:** Reviews Summary widget displayed on a web page.

**Reviews Summary** widget displaying the number of reviews posted to the web page.



## Links to Topics Widget Tag

The “Links to Topics Deployment” screen (**Reviews > Deploy > Links to Topics**) provides administrators and developers with the Links to Topics widget tag. This section analyzes the parameters defined in the Links to Topics widget tag:

### Links to Topics widget tag

```

1      <a href="http://host:port/test/page1.html#reviews_link
      &resource_id=ReviewsID"></a>
2      <div id="comments_link_div"></div>
3      <script type="text/javascript">
4          cos = window.cos || {};
5          cos.pageWidgets = cos.pageWidgets || [];

6          cos.pageWidgets.push({name: "comments-link",
7              version: "0.1",
8              elementID: "comments_link_div",
9              attributes: {"site_id":"FirstSiteII"}});

10         cos.pageScripts = cos.pageScripts || [];
11         cos.pageScripts.push('comments-link');

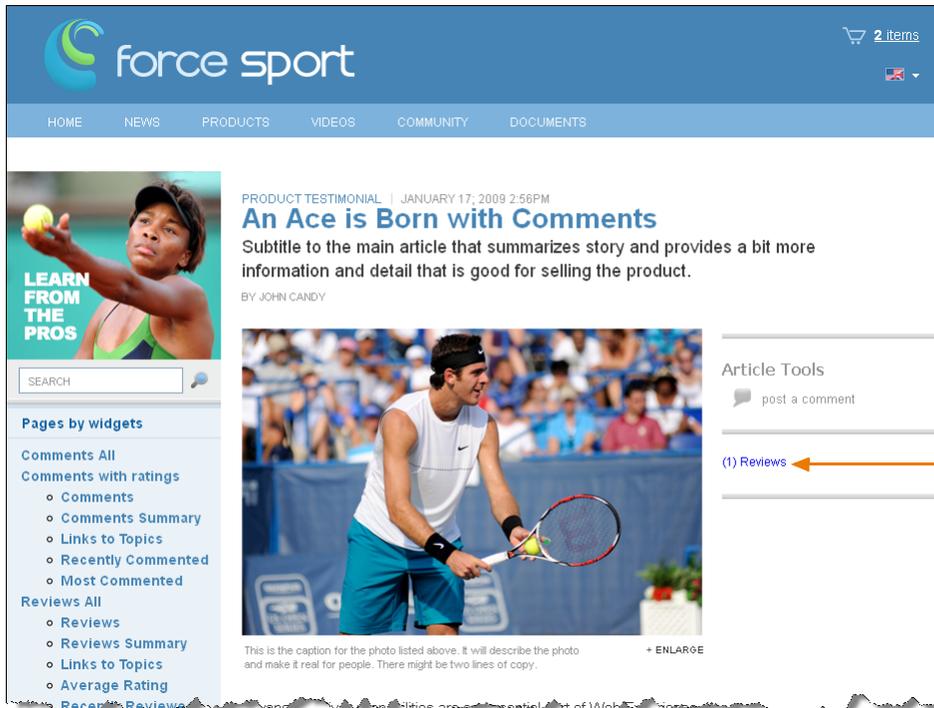
12         (function()
13         {
14             var oldOnloadHandler = window.onload || function()
15             {
16             };
17             if (!oldOnloadHandler.alreadyProcessed)
18             {
19                 window.onload = function()
20                 {
21                     var script = document.createElement
22                     ('script');
23                     script.src = 'http://cosservice.fatwire.com
24                         :8080/cos/wsdk/widget/'
25                         + cos.pageScripts.join(':') +
26                         '.js?site_id=FirstSiteII';
27                     script.type = 'text/javascript';
28                     script.charset = 'utf-8';
29                     document.getElementsByTagName("head")
30                     .item(0).appendChild(script);
31                     oldOnloadHandler.apply(this, arguments);
32                 };
33                 window.onload.alreadyProcessed = true;
34             }
35         })();
36     </script>

```

### Analyzing the Links to Topics widget tag

- [Line 1](#) is a link to the web page on which the Reviews widget tag is deployed, and must be added when the Links to Topics widget tag is inserted into the desired template.
- [Line 2](#) defines the container that holds the Links to Topics widget on the page.
- [Line 3](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 4](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 5](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 6 – 9](#) define the Links to Topics widget tag and add the new element to the list of widget tags (defined in [line 5](#)). [Line 8](#) contains a link that is used to render the Links to Topics widget inside the container defined in [line 2](#).
- [Line 10](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 11](#) adds the Links to Topics widget tag (`comments-link`) to the list of page scripts (defined in [line 10](#)).
- [Line 14](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic, then the function is left empty.
- [Line 17](#) checks if the `onload` function is already processed.
- [Line 19](#) overrides the code that runs when the page is loaded.
- [Lines 21 – 26](#) add the JavaScript tag to the page, which includes all the scripts that need to be deployed.
- [Line 27](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 29](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 32](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 9:** Links to Topics widget displayed on a web page



The **Links to Topics** widget displaying the number of reviews posted to a given web page. Click this link to view the reviews posted to the web page.

## Reviews Average Rating Widget Tag

The “Reviews Average Rating Deployment” screen (**Reviews > Deploy > Average Rating**) provides administrators and designers with the Reviews Average Rating widget tag. This section analyzes the parameters defined in the Reviews Average Rating widget tag:

### Reviews Average Rating widget tag

```

1     <div id="reviews_average_container_
      ReviewsWidgetResourceID"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.reviews_average",
6             version: "1.5",
7             elementID: "reviews_average_container_
8                 ReviewsWidgetResourceID",
9             attributes: {"site_id":"FirstSiteII","resource_id"
10                : "ReviewsWidgetResourceID"}});
11        setTimeout(
12            function(){if ((typeof(wsdk) ==
13                'undefined') || (typeof(wsdk.reviews_average) ==
14                'undefined'))
15                {document.getElementById('reviews_average_container_Reviews
16                WidgetResourceID').innerHTML = "<div style='font-family:
17                Tahoma, Verdana, Geneva, sans-serif;font-size: 12px;color:
18                #333333;border: 1px solid #dbdfef;padding-left:
19                5px;padding-top: 4px;height: 22px;'>Average Rating is
20                unavailable right now. Please contact the site
21                administrator.</div>";}}
22            ,30000);

23        cos.pageScripts = cos.pageScripts || [];
24        cos.pageScripts.push('wsdk.reviews_average');

25        (function()
26        {
27            var oldOnloadHandler = window.onload || function()
28            {
29            };
30            if (!oldOnloadHandler.alreadyProcessed)
31            {
32                window.onload = function()
33                {
34                    var script =
35                        document.createElement('script');
36                    script.src = 'http://cosservice.fatwire.com
37                        :8080/cos/wsdk/widget/'
38                        + cos.pageScripts.join(':') +
39                        '.js?site_id=FirstSiteII';
40                    script.type = 'text/javascript';

```

```

27         script.charset = 'utf-8';
28         document.getElementsByTagName( "head" )
           .item(0).appendChild(script);
29         oldOnloadHandler.apply(this, arguments);
30     };
31     window.onload.alreadyProcessed = true;
32 }

33     })();
34 </script>

```

### Analyzing the Reviews Average Rating widget tag

- **Line 1** defines the container that holds the Reviews Average Rating widget on the page. You must specify the resource ID of the Reviews widget that is deployed on the web page whose average rating the Reviews Average Rating widget should display. For example:

```
<div id="reviews_average_container_ReviewsWidgetResourceID">
  </div>
```

Where *ReviewsWidgetResourceID* is the resource ID of the desired Reviews widget.

- **Line 2** opens the bootstrapping JavaScript code that is needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Reviews Average Rating widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Reviews Average Rating widget inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Reviews Average Rating widget is rendered on the page. If the Reviews Average Rating widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- **Line 12** defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Reviews Average Rating widget tag (`wsdk.reviews_average`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add a JavaScript tag to the page, which includes a list of all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.

- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

### Notes for Deploying the Reviews Average Rating Widget Tag

To configure the Reviews Average Rating widget tag to display the average ratings of multiple web pages, you will have to insert the resource ID of the Reviews widget deployed on each desired web page into the Reviews Average Rating widget tag. For example:

1. In the menu bar, select **Reviews > Deploy > Average Rating**. In the “Reviews Average Rating Deployment” screen, copy (**Ctrl+C**) the widget tag.
2. Access the WebCenter Sites Admin interface and insert the Reviews Average Rating widget tag into the desired template. For instructions, see [step 4](#) in the section “[How Do I Deploy Community Widget Tags?](#)”
3. In the Reviews Average Rating widget tag, do the following:
  - a. Specify the resource ID of the Reviews widget deployed on each web page whose average rating the Reviews Average Rating widget should display. For each Reviews widget’s resource ID, add one of the following lines before [line 1](#) of the Reviews Average Rating widget tag:
    - If the Reviews widget is explicitly assigned a resource ID, add the following line:
 

```
<div id="reviews_ar&resource_id=ReviewsID"></div>
```

 where *ReviewsID* is the resource ID of the Reviews widget deployed on the web page whose average rating the Reviews Average Rating widget should display.
    - If the Reviews widget is not explicitly assigned a resource ID, add the following line:
 

```
<div id="reviews_ar&resource_id=http://host:port/demo/reviews.html"></div>
```

 where *http://host:port/demo/reviews.html* is the path to the web page on which the desired Reviews widget tag is deployed.
  - b. In [line 8](#), set the `resource_id` tag to `auto` (`"resource_id: "auto"`).

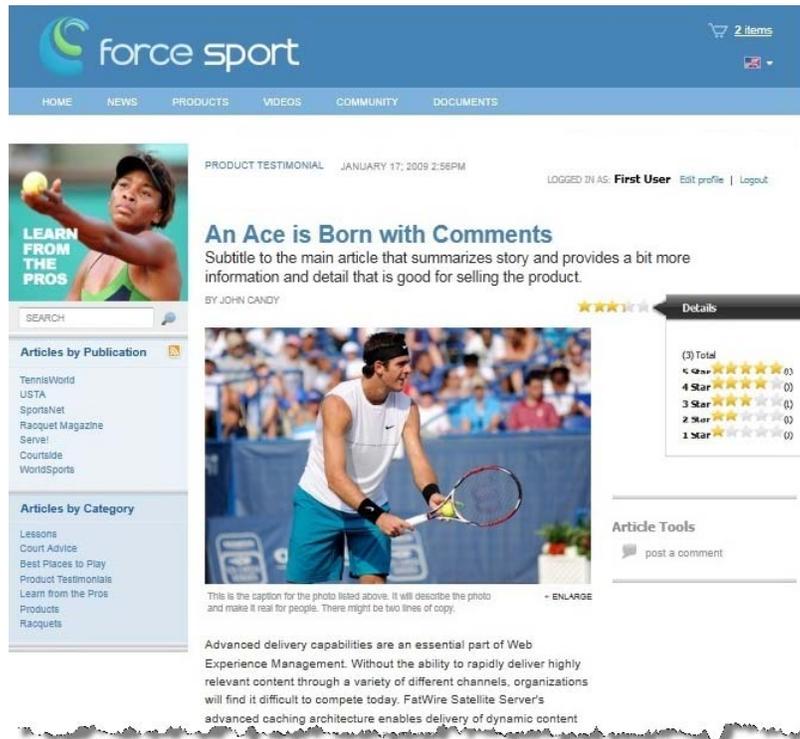
Your changes will look similar to the following code snippet:

```
<div id="reviews_ar&resource_id=ReviewsID"></div>
<div id="reviews_ar&resource_id=ReviewsID_1"></div>
<div id="reviews_ar&resource_id=ReviewsID_2"></div>
<div id="average_rating_container_ReviewsID"></div>
<script type="text/javascript">
  cos = window.cos || {};
  cos.pageWidgets = cos.pageWidgets || [];

  cos.pageWidgets.push({name: "average-rating",
    version: "0.1",
    elementID: "average_rating_container_ReviewsID",
    attributes:
    {"site_id": "FirstSiteII", "resource_id": "auto", "type": "reviews"}});
  ...
</script>
```

4. Click **Save Changes**.
5. Preview the page and then publish the template to the website. For instructions, see [steps 6 – 8](#) on [page 31](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

**Figure 10:** Reviews Average Rating widget displayed on a web page



**Reviews Average Rating** widget displaying the average rating of a single web page. Point to the average rating field to render the "Details" pop-up box.



**Reviews Average Rating** widget displaying the average rating of three different web pages. Point to one of the average ratings to render the "Detail" pop-up box for the corresponding page.

## Top Ranked Reviews Widget Tag

The “Top Ranked Reviews Deployment” screen (**Reviews > Deploy > Top Ranked Topics**) provides administrators and designers with the Top Ranked Reviews widget tag. This section analyzes the parameters defined in the Top Ranked Topics widget tag:

### Top Ranked Reviews widget tag

```

1     <div id="topics_container_top_ranked_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5     cos.pageWidgets.push({name: "wsdk.topics",
6         version: "1.5",
7         elementID: "_top_ranked_other_container",
8         attributes: {"content_type":"top_ranked",
9             "resource_type":"other","site_id":"FirstSiteII","count
10            _topics":"10"}});
11     setTimeout(
12         function(){if ((typeof(wsdk) ==
13         'undefined') || (typeof(wsdk.topics) == 'undefined'))
14         {document.getElementById('topics_container_top_ranked
15         _other').innerHTML = "<div style='font-family: Tahoma,
16         Verdana, Geneva, sans-serif;font-size: 12px;color:
17         #333333;border: 1px solid #dbdfef;padding-left:
18         5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19         right now. Please contact the site administrator.</div>";}}
20         ,30000);

21     cos.pageScripts = cos.pageScripts || [];
22     cos.pageScripts.push('wsdk.topics');

23     (function()
24     {
25         var oldOnloadHandler = window.onload || function()
26         {
27             };
28         if (!oldOnloadHandler.alreadyProcessed)
29         {
30             window.onload = function()
31             {
32                 var script = document.createElement
33                 ('script');
34                 script.src = 'http://cosservice.fatwire.com
35                 :8080/cos/wsdk/widget/'
36                 + cos.pageScripts.join(':') +
37                 '.js?site_id=FirstSiteII';
38                 script.type = 'text/javascript';
39                 script.charset = 'utf-8';
40                 document.getElementsByTagName("head")
41                 .item(0).appendChild(script);
42                 oldOnloadHandler.apply(this, arguments);

```

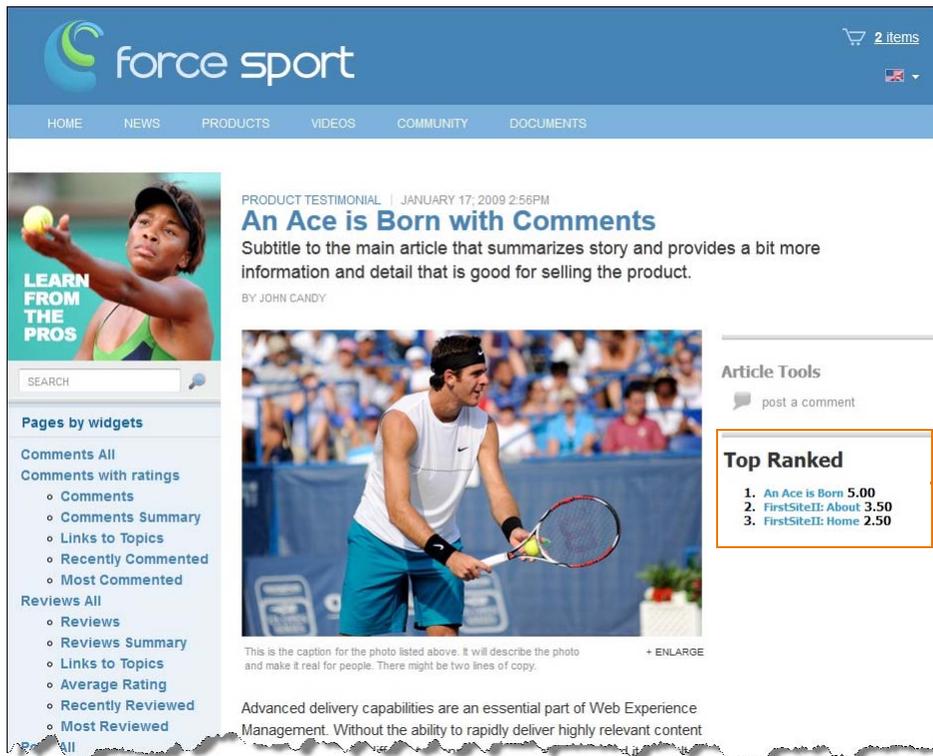
```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Top Ranked Reviews widget tag

- [Line 1](#) defines the container that holds the Top Ranked Reviews widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Top Ranked Reviews widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Top Ranked Reviews widget inside the container defined in [line 1](#). [Line 8](#) specifies the Top Ranked Reviews widget tag's settings, including the type of topics the Top Ranked Reviews widget will list. If you want the Top Ranked Reviews widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Top Ranked Reviews widget is rendered on the page. If the Top Ranked Reviews widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Top Ranked Reviews widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 11:** Top Ranked Reviews widget displayed on a web page



**Top Ranked Reviews** widget displaying the names of the highest rated web pages and the average rating of each page.

## Recently Reviewed Widget Tag

The “Recently Reviewed Deployment” screen (**Reviews > Deploy > Recently Reviewed**) provides administrators and designers with the Recently Reviewed widget tag. This section analyzes the parameters defined in the Recently Reviewed widget tag:

### Recently Reviewed widget tag

```

1     <div id="topics_container_recently_reviewed_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wSDK.topics",
6             version: "1.5",
7             elementID: "_recently_reviewed_other_container",
8             attributes: {"content_type": "recently_reviewed",
9                 "resource_type": "other", "site_id":
10                  "FirstSiteII", "count_topics": "10"}});
11         setTimeout(
12             function(){if ((typeof(wSDK) ==
13                 'undefined') || (typeof(wSDK.topics) == 'undefined'))
14                 {document.getElementById('topics_container_recently
15                 _reviewed_other').innerHTML = "<div style='font-family:
16                 Tahoma, Verdana, Geneva, sans-serif;font-size: 12px;color:
17                 #333333;border: 1px solid #dbdfef;padding-left:
18                 5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19                 right now. Please contact the site administrator.</div>";}}
20             ,30000);

21         cos.pageScripts = cos.pageScripts || [];
22         cos.pageScripts.push('wSDK.topics');

23         (function()
24         {
25             var oldOnloadHandler = window.onload || function()
26             {
27                 };
28             if (!oldOnloadHandler.alreadyProcessed)
29             {
30                 window.onload = function()
31                 {
32                     var script = document.createElement
33                     ('script');
34                     script.src = 'http://coservice.fatwire.com
35                     :8080/cos/wSDK/widget/'
36                     + cos.pageScripts.join(':') +
37                     '.js?site_id=FirstSiteII';
38                     script.type = 'text/javascript';
39                     script.charset = 'utf-8';
40                     document.getElementsByTagName("head")
41                     .item(0).appendChild(script);
42                     oldOnloadHandler.apply(this, arguments);

```

```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Recently Reviewed widget tag

- [Line 1](#) defines the container that holds the Recently Reviewed widget tag on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widget tags deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Recently Reviewed widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Recently Reviewed widget inside the container defined in [line 1](#). [Line 8](#) specifies the Recently Reviewed widget tag's settings, including the type of topics the Recently Reviewed widget will list. If you want the Recently Reviewed widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Recently Reviewed widget is rendered on the page. If the Recently Reviewed widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Recently Reviewed widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag to the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 12:** Recently Reviewed widget displayed on a web page

The screenshot shows a Force Sport website with a main article titled "An Ace is Born with Comments" and a "Recently Reviewed" widget. The widget lists three topics: "An Ace is Born", "Tennis World", and "Player Information". Annotations point to the widget title and the list of topics.

**Annotations:**

- Title for this widget
- Recently Reviewed widget listing the topics to which site visitors have recently posted reviews. Click a topic's name to view the topic on the website.

**Note**

When the Recently Reviewed widget is deployed, it does not display a title (as shown in [Figure 12](#)). For the Recently Reviewed widget to display a title, you will have to insert the title into the template to which you copied the Recently Reviewed widget tag.

## Most Reviewed Widget Tag

The “Most Reviewed Deployment” screen (**Reviews > Deploy > Most Reviewed**) provides administrators and designers with the Most Reviewed widget tag. This section analyzes the parameters defined in the Most Reviewed widget tag:

### Most Reviewed widget tag

```

1     <div id="topics_container_most_reviewed_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.topics",
6             version: "1.5",
7             elementID: "_most_reviewed_other_container",
8             attributes: {"content_type":"most_reviewed",
9                 "resource_type":"other","site_id":
10                "FirstSiteII","count_topics":"10"}});
11        setTimeout(
12            function(){if ((typeof(wsdk) ==
13                'undefined') || (typeof(wsdk.topics) == 'undefined'))
14                {document.getElementById('topics_container_most_reviewed
15                _other').innerHTML = "<div style='font-family: Tahoma,
16                Verdana, Geneva, sans-serif;font-size: 12px;color:
17                #333333;border: 1px solid #dbdfef;padding-left:
18                5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19                right now. Please contact the site administrator.</div>";}}
20            ,30000);

21        cos.pageScripts = cos.pageScripts || [];
22        cos.pageScripts.push('wsdk.topics');

23        (function()
24        {
25            var oldOnloadHandler = window.onload || function()
26            {
27                };
28            if (!oldOnloadHandler.alreadyProcessed)
29            {
30                window.onload = function()
31                {
32                    var script = document.createElement
33                    ('script');
34                    script.src = 'http://cosservice.fatwire.com
35                    :8080/cos/wsdk/widget/'
36                    + cos.pageScripts.join(':') +
37                    '.js?site_id=FirstSiteII';
38                    script.type = 'text/javascript';
39                    script.charset = 'utf-8';
40                    document.getElementsByTagName("head")
41                    .item(0).appendChild(script);
42                    oldOnloadHandler.apply(this, arguments);

```

```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Most Reviewed widget tag

- [Line 1](#) defines the container that holds the Most Reviewed widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Most Reviewed widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Most Reviewed widget inside the container defined in [line 1](#). [Line 8](#) specifies the Most Reviewed widget tag's settings, including the type of topics the Most Reviewed widget will list. If you want the Most Reviewed widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Most Reviewed widget is rendered on the page. If the Most Reviewed widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no such scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Most Reviewed widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag to the page, including a list of scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 13:** Most Reviewed widget displayed on a web page

The screenshot shows a web page for 'force sport'. The main article is titled 'An Ace is Born with Comments' and includes a subtitle: 'Subtitle to the main article that summarizes story and provides a bit more information and detail that is good for selling the product.' The article is by John Candy and dated January 17, 2009. A 'Most Reviewed' widget is positioned to the right of the article, listing three items: 'An Ace is Born', 'Tennis World', and 'Player Information'. The widget title 'Most Reviewed' is highlighted with an orange box. Annotations with arrows point to the title and the list items, explaining their function.

**Note**

When the Most Reviewed widget is deployed, it does not display a title (as shown in [Figure 13](#)). For the Most Reviewed widget to display a title, you will have to insert the title into the template to which you copied the Most Reviewed widget tag.

## Ratings Widget Tags

The widget tags related to adding rating functionality to your website are the following:

- [Stars Ratings Widget Tag](#)
- [Thumbs Up/Down Ratings Widget Tag](#)
- [Like It Ratings Widget Tag](#)
- [Recommend Ratings Widget Tag](#)
- [Ratings Average Rating Widget Tag](#)
- [Recently Rated Widget Tag](#)
- [Most Rated Widget Tag](#)

### Stars Ratings Widget Tag

The “Stars Deployment” screen (**Ratings > Deploy > Stars Ratings**) provides administrators and designers with the Stars Ratings widget tag. This section analyzes the parameters defined in the Stars Ratings widget tag:

#### Stars Ratings widget tag

```

1     <div id="ratings_stars_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.ratings",
6             version: "1.5",
7             elementID: "ratings_stars_container",
8             attributes:
9             {"rating_type":"stars","site_id":"FirstSiteII"}});
10        setTimeout(
11            function(){if ((typeof(wsdk) ==
12                'undefined') || (typeof(wsdk.ratings) == 'undefined'))
13                {document.getElementById('ratings_stars_container').innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
14                sans-serif;font-size: 12px;color: #333333;border: 1px solid
15                #dbdfef;padding-left: 5px;padding-top: 4px;height:
16                22px;'>Ratings is unavailable right now. Please contact the
17                site administrator.</div>";}}
18            ,30000);

19        cos.pageScripts = cos.pageScripts || [];
20        cos.pageScripts.push('wsdk.ratings');

21        (function()
22        {
23            var oldOnloadHandler = window.onload || function()
24            {
25            };
26            if (!oldOnloadHandler.alreadyProcessed)
27            {

```

```

21         window.onload = function()
22         {
23             var script = document.createElement
24                 ('script');
25             script.src = 'http://coservice.fatwire.com
26                 :8080/cos/wsdk/widget/'
27                 + cos.pageScripts.join(':') +
28                 '.js?site_id=FirstSiteII';
29             script.type = 'text/javascript';
30             script.charset = 'utf-8';
31             document.getElementsByTagName("head")
32                 .item(0).appendChild(script);
33             oldOnloadHandler.apply(this, arguments);
34         };
35     window.onload.alreadyProcessed = true;
36 }
37
38     })();
39 </script>

```

### Analyzing the Stars Ratings widget tag

- **Line 1** defines the container that holds the Stars Ratings widget on the page. If you assign a resource ID to this instance of the Stars Ratings widget tag, the widget's resource ID is also specified in this line. For example:

```
<div id="ratings_stars_container_StarsRatingsResourceID_
    other"></div>
```

Where *StarsRatingsResourceID* is the resource ID assigned to this instance of the Stars Ratings widget.

**Note:** If you do not specify a resource ID for this widget tag, the encrypted URL of the page on which this widget tag is deployed is assigned as the widget's resource ID.

- **Line 2** opens the bootstrapping JavaScript code needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Stars Ratings widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Stars Ratings widget inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Stars Ratings widget is rendered on the page. If the Stars Ratings widget is not rendered after 30 seconds, users are informed that there is an error and that they should contact the site administrator.
- **Line 12** defines the list of scripts that are deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Stars Ratings widget tag (`wsdk.ratings`) to the list of page scripts (defined in **line 12**).

- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- **Line 34** closes the bootstrapping JavaScript code needed for the widget.

**Figure 14:** Stars Ratings widget deployed on a web page



**Stars Ratings**  
widget. Rate the  
topic by clicking the  
desired number of  
stars.

force sport

HOME NEWS PRODUCTS VIDEOS COMMUNITY DOCUMENTS

LEARN FROM THE PROS

PRODUCT TESTIMONIAL | JANUARY 17, 2009 2:56PM  
**An Ace is Born with Comments**  
Subtitle to the main article that summarizes story and provides a bit more information and detail that is good for selling the product.  
BY JOHN CANDY

SEARCH

Pages by widgets

Comments All  
Comments with ratings

- o Comments
- o Comments Summary
- o Links to Topics
- o Recently Commented
- o Most Commented

Reviews All

- o Reviews
- o Reviews Summary
- o Links to Topics
- o Average Rating
- o Recently Reviewed
- o Most Reviewed

Polls All

- o Poll
- o Result Poll

Ratings All

- o Ratings
- o Average Rating
- o Resently Rated
- o Most Rated

This is the caption for the photo listed above. It will describe the photo and make it real for people. There might be two lines of copy.

Advanced delivery capabilities are an essential part of Web Experience Management. Without the ability to rapidly deliver highly relevant content through a variety of different channels, organizations will find it difficult to compete today. FatWire Satellite Server's advanced caching architecture enables delivery of dynamic content on an enterprise scale. Meanwhile, FatWire Mobility Server allows seamless integration of mobile and traditional Web channels.

Learn how FatWire Satellite Server and [Mobility Server](#) can ensure your dynamic web presence maintains peak performance even during times of peak demand. Satellite Server caches static elements of site pages on production servers for rapid-fire delivery.

Rate this: ★★★★★

## Thumbs Up/Down Ratings Widget Tag

The “Thumbs Up/Down Ratings Deployment” screen (**Ratings > Deploy > Thumbs Up/Down Ratings**) provides administrators and designers with the Thumbs Up/Down Ratings widget tag. This section analyzes the parameters defined in the Thumbs Up/Down Ratings widget tag:

### Thumbs Up/Down Ratings widget tag

```

1     <div id="ratings_thumbs_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.ratings",
6             version: "1.5",
7             elementID: "ratings_thumbs_container",
8             attributes:
9             {"rating_type":"thumbs","site_id":"FirstSiteII"}});
10        setTimeout(
11            function(){if ((typeof(wsdk) ==
12                'undefined') || (typeof(wsdk.ratings) == 'undefined'))
13                {document.getElementById('ratings_thumbs_container').innerHTML
14                TML = "<div style='font-family: Tahoma, Verdana, Geneva,
15                sans-serif;font-size: 12px;color: #333333;border: 1px solid
16                #dbdfef;padding-left: 5px;padding-top: 4px;height:
17                22px;'>Ratings is unavailable right now. Please contact the
18                site administrator.</div>";}}
19            ,30000);

20        cos.pageScripts = cos.pageScripts || [];
21        cos.pageScripts.push('wsdk.ratings');

22        (function()
23        {
24            var oldOnloadHandler = window.onload || function()
25            {
26                };
27            if (!oldOnloadHandler.alreadyProcessed)
28            {
29                window.onload = function()
30                {
31                    var script = document.createElement
32                    ('script');
33                    script.src = 'http://cosservice.fatwire.com
34                    :8080/cos/wsdk/widget/'
35                    + cos.pageScripts.join(':') +
36                    '.js?site_id=FirstSiteII';
37                    script.type = 'text/javascript';
38                    script.charset = 'utf-8';
39                    document.getElementsByTagName("head")
40                    .item(0).appendChild(script);
41                    oldOnloadHandler.apply(this, arguments);

```

```

30             };
31             window.onload.alreadyProcessed = true;
32         }

33     })();
34 </script>

```

### Analyzing the Thumbs Up/Down Ratings widget tag

- [Line 1](#) defines the container that holds the Thumbs Up/Down Ratings widget on the page. If you assign a resource ID to this instance of the Thumbs Up/Down Ratings widget, the resource ID is also specified in this line. For example:

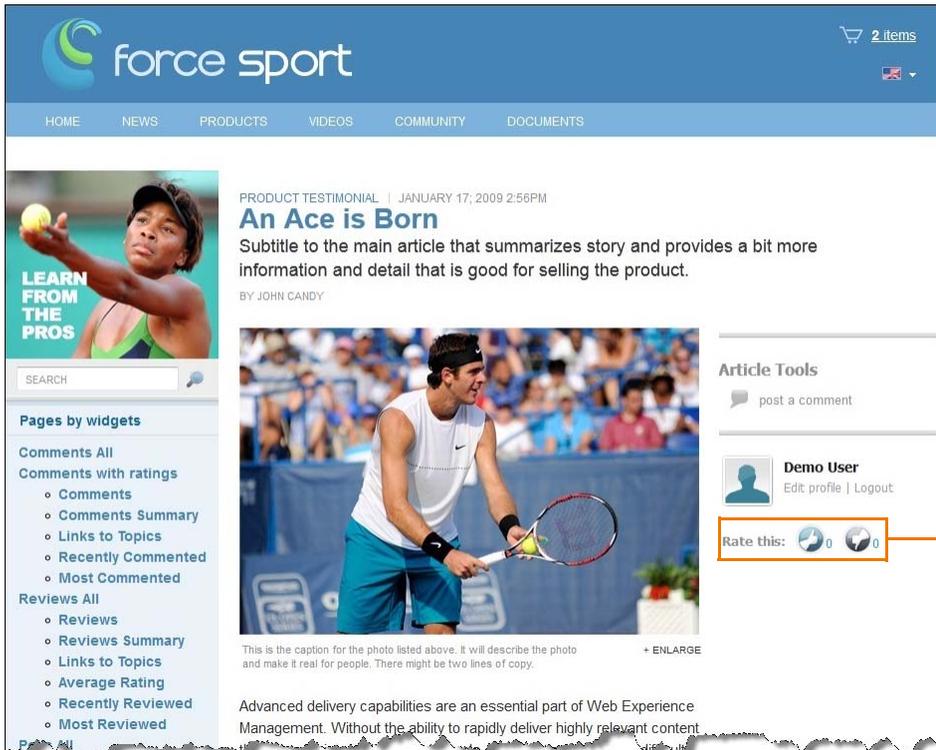
```
<div id="ratings_thumbs_container_ThumbsRatingsResourceID_
    other"></div>
```

Where *ThumbsRatingsResourceID* is the resource ID assigned to this instance of the Thumbs Up/Down Ratings widget.

**Note:** If you do not specify a resource ID for this widget tag, the encrypted URL of the page on which this widget tag is deployed is assigned as the widget's resource ID.

- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Thumbs Up/Down Ratings widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Thumbs Up/Down Ratings widget inside the container defined in [line 1](#).
- [Lines 9 – 11](#) check if the Thumbs Up/Down Ratings widget is rendered on the page. If the Thumbs Up/Down Ratings widget is not rendered after 30 seconds, users are informed that there is an error and that they should contact the site administrator.
- [Line 12](#) defines the list of scripts that are deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Thumbs Up/Down Ratings widget tag (`wsdk.ratings`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

Figure 15: Thumbs Up/Down Ratings widget displayed on a web page



Thumbs Up/Down Ratings widget displaying a field which visitors can use to indicate whether the topic is informative.

## Like It Ratings Widget Tag

The “Like It Deployment” screen (**Ratings > Deploy > Like It Ratings**) provides administrators and designers with the Like It Ratings widget tag. This section analyzes the parameters defined in the Like It Ratings widget tag:

### Like It Ratings widget tag

```
1     <div id="ratings_like_it_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.ratings",
6             version: "1.5",
7             elementID: "ratings_like_it_container",
8             attributes:
9             {"rating_type":"like_it","site_id":"FirstSiteII"}});
10        setTimeout(
11            function(){if ((typeof(wsdk) ==
12                'undefined') || (typeof(wsdk.ratings) == 'undefined'))
13                {document.getElementById('ratings_like_it_container').innerHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
14                sans-serif;font-size: 12px;color: #333333;border: 1px solid
15                #dbdfef;padding-left: 5px;padding-top: 4px;height:
16                22px;'>Ratings is unavailable right now. Please contact the
17                site administrator.</div>";}}
18                ,30000);

19        cos.pageScripts = cos.pageScripts || [];
20        cos.pageScripts.push('wsdk.ratings');

21        (function()
22        {
23            var oldOnloadHandler = window.onload || function()
24            {
25                };
26            if (!oldOnloadHandler.alreadyProcessed)
27            {
28                window.onload = function()
29                {
30                    var script = document.createElement
31                        ('script');
32                    script.src = 'http://cosservice.fatwire.com
33                        :8080/cos/wsdk/widget/'
34                        + cos.pageScripts.join(':') +
35                        '.js?site_id=FirstSiteII';
36                    script.type = 'text/javascript';
37                    script.charset = 'utf-8';
38                    document.getElementsByTagName("head")
39                        .item(0).appendChild(script);
40                    oldOnloadHandler.apply(this, arguments);
41                }
42            }
43        });
```

```
31         window.onload.alreadyProcessed = true;
32     }
33     }));
34 </script>
```

### Analyzing the Like It Ratings widget tag

- **Line 1** defines the container that holds the Like It Ratings widget on the page. If you assign a resource ID to this instance of the Like It Ratings widget, the widget's resource ID is also specified in this line. For example:

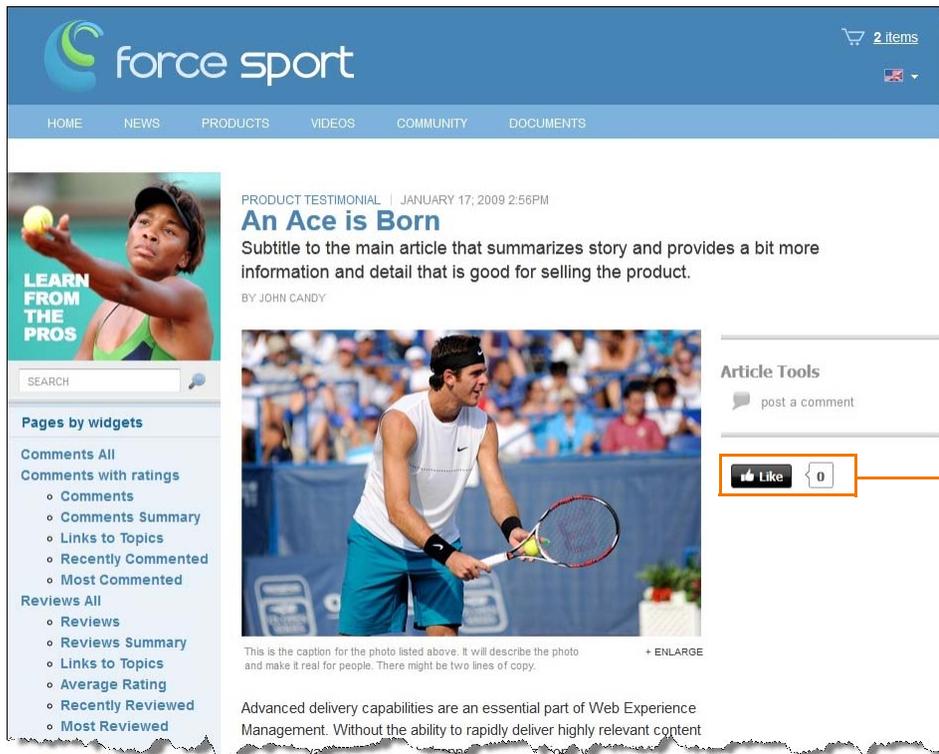
```
<div id="ratings_like_it_container_LikeItRatingsResourceID_
    other"></div>
```

Where *LikeItRatingsResourceID* is the resource ID assigned to this instance of the Like It Ratings widget.

**Note:** If you do not specify a resource ID for this widget tag, the encrypted URL of the page on which this widget tag is deployed is assigned as the widget's resource ID.

- **Line 2** opens the bootstrapping JavaScript code needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Like It Ratings widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Like It Ratings widget tag inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Like It Ratings widget is rendered on the page. If the Like It Ratings widget is not rendered after 30 seconds, users are informed that there is an error and that they should contact the site administrator.
- **Line 12** defines the list of scripts that are deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Like It Ratings widget tag (`wsdk.ratings`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- **Line 34** closes the bootstrapping JavaScript code needed for the widget.

Figure 16: Like It Ratings widget displayed on a web page



Like it Ratings widget displaying a Like link which visitors can click to indicate whether they liked a given topic.

## Recommend Ratings Widget Tag

The “Recommend Deployment” screen (**Ratings > Deploy > Recommend Ratings**) provides administrators and designers with the Recommend Ratings widget tag. This section analyzes the parameters defined in the Recommend Ratings widget tag:

### Recommend Ratings widget tag

```

1   <div id="ratings_recommend_container"></div>
2   <script type="text/javascript">
3       cos = window.cos || {};
4       cos.pageWidgets = cos.pageWidgets || [];

5       cos.pageWidgets.push({name: "wsdk.ratings",
6           version: "1.5",
7           elementID: "ratings_recommend_container",
8           attributes:
9       {"rating_type":"recommend","site_id":"FirstSiteII"}});
10      setTimeout(
11          function(){if ((typeof(wsdk) ==
12      'undefined') || (typeof(wsdk.ratings) == 'undefined'))
13      {document.getElementById('ratings_recommend_container').inn
14      erHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
15      sans-serif;font-size: 12px;color: #333333;border: 1px solid
16      #dbdfef;padding-left: 5px;padding-top: 4px;height:
17      22px;'>Ratings is unavailable right now. Please contact the
18      site administrator.</div>";}}
19      ,30000);

20      cos.pageScripts = cos.pageScripts || [];
21      cos.pageScripts.push('wsdk.ratings');

22      (function()
23      {
24          var oldOnloadHandler = window.onload || function()
25          {
26              };
27          if (!oldOnloadHandler.alreadyProcessed)
28          {
29              window.onload = function()
30              {
31                  var script = document.createElement
32                  ('script');
33                  script.src = 'http://cosservice.fatwire.com
34                  :8080/cos/wsdk/widget/'
35                  + cos.pageScripts.join(':') +
36                  '.js?site_id=FirstSiteII';
37                  script.type = 'text/javascript';
38                  script.charset = 'utf-8';
39                  document.getElementsByTagName("head")
40                  .item(0).appendChild(script);
41                  oldOnloadHandler.apply(this, arguments);
42              }
43          }
44      });

```

```
31         window.onload.alreadyProcessed = true;
32     }
33     }));
34 </script>
```

### Analyzing the Recommend Ratings widget tag

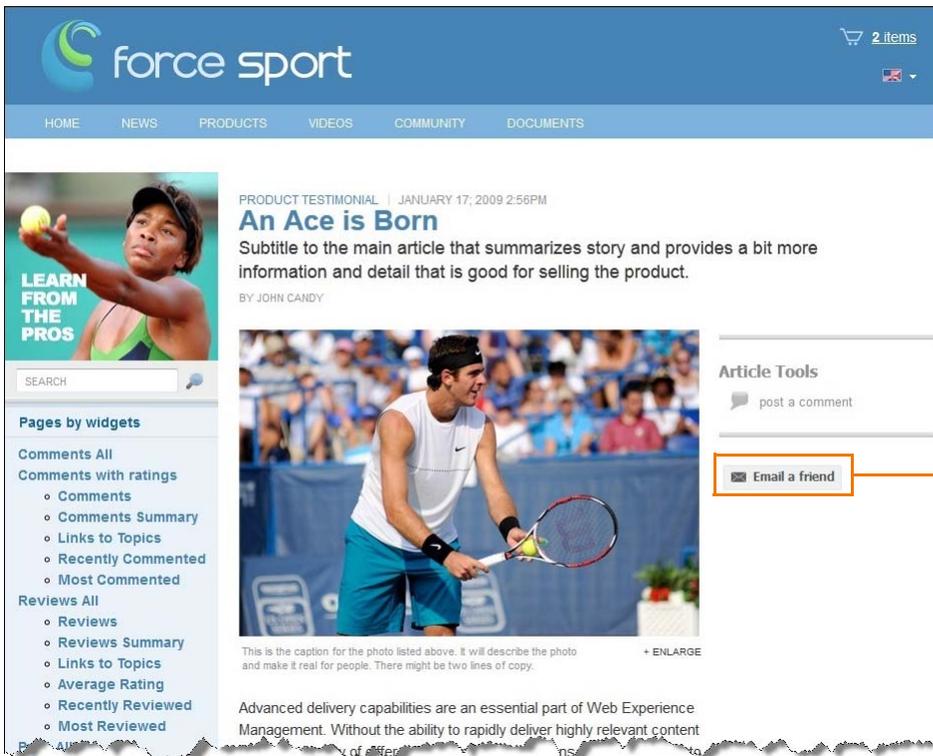
- **Line 1** defines the container that holds the Recommend Ratings widget on the page. If you assign a resource ID to this instance of the Recommend Ratings widget, the widget's resource ID is also specified in this line. For example:

```
<div id="ratings_recommend_container_RecommendRatingsResourceID
    _other"></div>
```

Where *RecommendRatingsResourceID* is the resource ID assigned to this instance of the Recommend Ratings widget.

**Note:** If you do not specify a resource ID for this widget tag, the encrypted URL of the page on which this widget tag is deployed is assigned as the widget's resource ID.

- **Line 2** opens the bootstrapping JavaScript code needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Recommend Ratings widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Recommend Ratings widget inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Recommend Ratings widget is rendered on the page. If the Recommend Ratings widget is not rendered after 30 seconds, users are informed that there is an error and that they should contact the site administrator.
- **Line 12** defines the list of scripts that are deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Recommend Ratings widget tag (`wsdk.ratings`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- **Line 34** closes the bootstrapping JavaScript code needed for the widget.

**Figure 17:** Recommend Ratings widget displayed on a web page

**Recommend Ratings** widget displaying an **Email a Friend** link which visitors can click to email the URL of the web page to a contact who may find the content informative.

## Ratings Average Rating Widget Tag

The “Ratings Average Rating Deployment” screen (**Ratings > Deploy > Average Rating**) provides administrators and designers with the Ratings Average Rating widget tag. This section analyzes the parameters defined in the Ratings Average Rating widget tag:

### Ratings Average Rating widget tag

```

1     <div id="ratings_average_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.ratings_average",
6             version: "1.5",
7             elementID: "ratings_average_container",
8             attributes: {"site_id":"FirstSiteII"}});
9         setTimeout(
10            function(){if ((typeof(wsdk) ==
'undefined') || (typeof(wsdk.ratings_average) ==
'undefined'))
{document.getElementById('ratings_average_container').inner
HTML = "<div style='font-family: Tahoma, Verdana, Geneva,
sans-serif;font-size: 12px;color: #333333;border: 1px solid
#dbdfef;padding-left: 5px;padding-top: 4px;height:
22px;'>Average Rating is unavailable right now. Please
contact the site administrator.</div>";}}
11            ,30000);

12         cos.pageScripts = cos.pageScripts || [];
13         cos.pageScripts.push('wsdk.ratings_average');

14         (function()
15         {
16             var oldOnloadHandler = window.onload || function()
17             {
18             };
19             if (!oldOnloadHandler.alreadyProcessed)
20             {
21                 window.onload = function()
22                 {
23                     var script = document.createElement
24                     ('script');
25                     script.src = 'http://cosservice.fatwire.com
:8080/cos/wsdk/widget/'
26                         + cos.pageScripts.join(':') +
27                         '.js?site_id=FirstSiteII';
28                     script.type = 'text/javascript';
29                     script.charset = 'utf-8';
30                     document.getElementsByTagName("head")
.item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
};

```

```
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Ratings Average Rating widget tag

- **Line 1** defines the container that holds the Ratings Average Rating widget on the page. You must specify the resource ID of the Stars Ratings widget tag deployed on the web page whose average rating the Ratings Average Ratings widget should display. For example:

```
<div id="ratings_average_container_StarsRatingsResourceID_
    other"></div>
```

Where *StarsRatingsResourceID* is the resource ID of the Stars Ratings widget deployed on the web page whose average rating the Ratings Average Rating widget should display.

- **Line 2** opens the bootstrapping JavaScript code that is needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Ratings Average Rating widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Ratings Average Rating widget tag inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Ratings Average Ratings widget is rendered on the page. If the Ratings Average Rating widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- **Line 12** defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Ratings Average Rating widget tag (`wsdk.ratings_average`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add the JavaScript tag to the page, which includes a list of all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- **Line 34** closes the bootstrapping JavaScript code needed for the widget.

### Notes for Deploying the Ratings Average Rating Widget Tag

To configure the Ratings Average Rating widget to display the average rating of multiple web pages, you will have to insert the resource ID of the Stars Ratings widget deployed on each desired web page into the Ratings Average Rating widget tag. For example:

1. In the menu bar, select **Ratings > Deploy > Average Rating**. In the “Ratings Average Rating Deployment” screen, copy (**Ctrl+C**) the widget tag.
2. Access the WebCenter Sites Admin interface and insert the Ratings Average Rating widget tag into the desired template. For instructions, see [step 4](#) in the section “[How Do I Deploy Community Widget Tags?](#)”
3. In the Ratings Average Rating widget tag, do the following:

- a. Specify the resource ID of the Stars Ratings widget deployed on each web page whose average rating the Ratings Average Rating widget should display. For each Stars Ratings widget’s resource ID, add one of the following lines before [line 1](#) of the Ratings Average Rating widget tag:

- If the Stars Ratings widget is explicitly assigned a resource ID, add the following line:

```
<div id="ratings_ar&resource_id=StarsRatingsID"></div>
```

where *StarsRatingsID* is the resource ID of the Stars Ratings widget deployed on the web page whose average rating the Ratings Average Rating widget should display.

- If the Stars Ratings widget is not explicitly assigned a resource ID, add the following line:

```
<div id="ratings_ar&resource_id=http://host:port/demo/Starsratings.html">
</div>
```

where *http://host:port/demo/Starsratings.html* is the path to the web page on which the Stars Ratings widget tag is deployed.

- b. In [line 8](#), set the `resource_id` tag to `auto` (`"resource_id:auto"`).

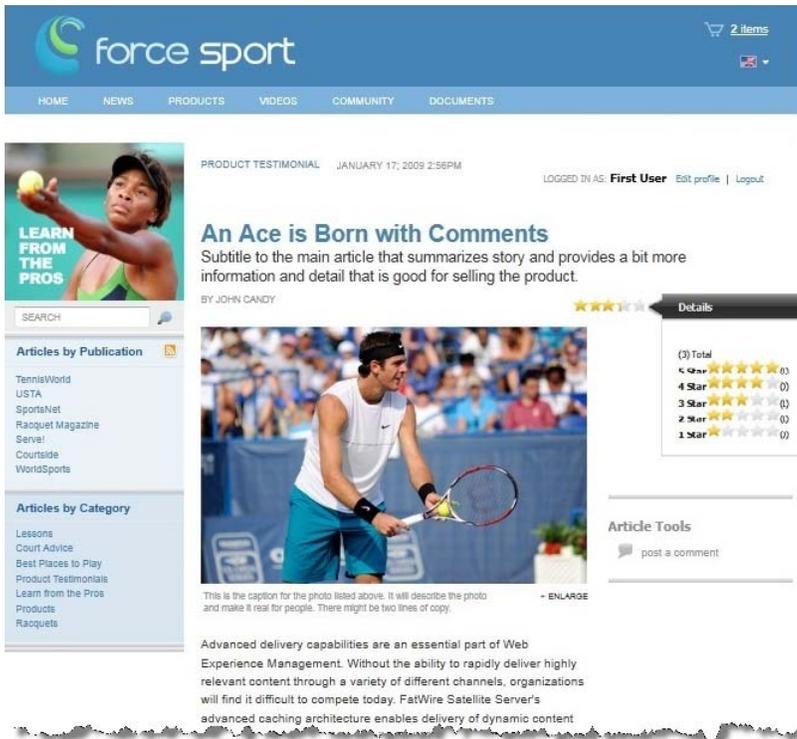
Your changes will look similar to the following:

```
<div id="reviews_ar&resource_id=RatingsID"></div>
<div id="reviews_ar&resource_id=RatingsID_1"></div>
<div id="reviews_ar&resource_id=RatingsID_2"></div>
<div id="average_rating_container_RatingsID"></div>
<script type="text/javascript">
  cos = window.cos || {};
  cos.pageWidgets = cos.pageWidgets || [];

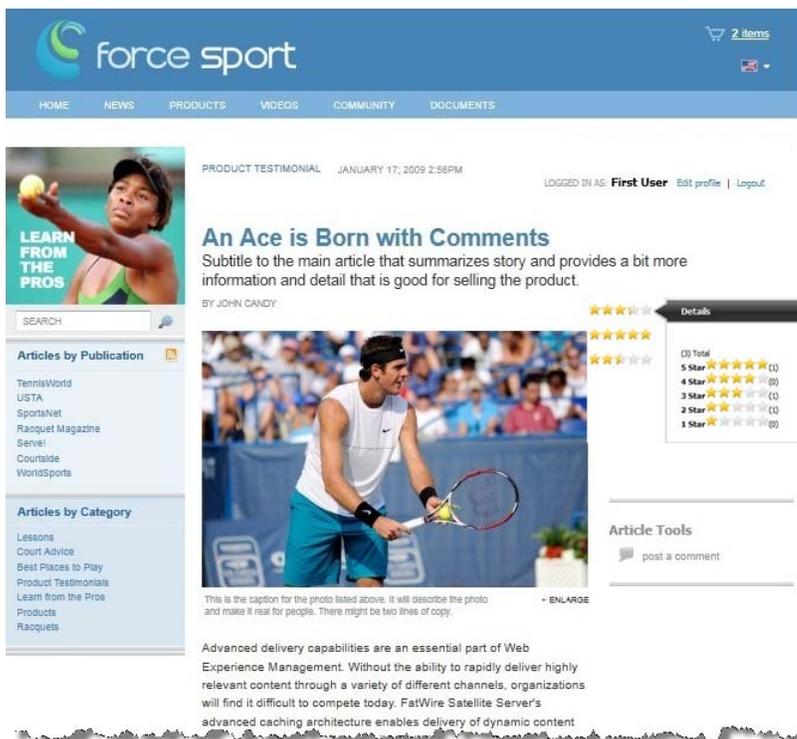
  cos.pageWidgets.push({name: "average-rating",
    version: "0.1",
    elementID: "average_rating_container_RatingsID",
    attributes:
    {"site_id":"FirstSiteII", "resource_id":"auto", "type": "ratings"}});
  ...
```

4. Click **Save Changes**.
5. Preview the page and then publish the template to the website. For instructions, see [steps 6 – 8](#) on [page 31](#) in the section “[How Do I Deploy Community Widget Tags?](#)”

Figure 18: Ratings Average Rating widget displayed on a web page



**Ratings Average Rating** widget displaying the average rating of a single web page. Point to the average rating to render the "Detail" pop-up box.



**Ratings Average Rating** widget displaying the average rating of three different web pages. Point to one of the average ratings to render the "Detail" pop-up box for the corresponding web page.

## Recently Rated Widget Tag

The “Recently Rated Deployment” screen (**Ratings > Deploy > Recently Rated**) provides administrators and designers with the Recently Rated widget tag. This section analyzes the parameters defined in the Recently Rated widget tag:

### Recently Rated widget tag

```

1     <div id="topics_container_recently_rated_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.topics",
6             version: "1.5",
7             elementID: "_recently_rated_other_container",
8             attributes: {"content_type": "recently_rated",
9                 "resource_type": "other", "site_id":
10                "FirstSiteII", "count_topics": "10"}});
11        setTimeout(
12            function(){if ((typeof(wsdk) ==
13                'undefined') || (typeof(wsdk.topics) == 'undefined'))
14                {document.getElementById('topics_container_recently_rated
15                _other').innerHTML = "<div style='font-family: Tahoma,
16                Verdana, Geneva, sans-serif;font-size: 12px;color:
17                #333333;border: 1px solid #dbdfef;padding-left:
18                5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19                right now. Please contact the site administrator.</div>";}}
20            ,30000);

21        cos.pageScripts = cos.pageScripts || [];
22        cos.pageScripts.push('wsdk.topics');

23        (function()
24        {
25            var oldOnloadHandler = window.onload || function()
26            {
27                };
28            if (!oldOnloadHandler.alreadyProcessed)
29            {
30                window.onload = function()
31                {
32                    var script = document.createElement
33                    ('script');
34                    script.src = 'http://cosservice.fatwire.com
35                    :8080/cos/wsdk/widget/'
36                    + cos.pageScripts.join(':') +
37                    '.js?site_id=FirstSiteII';
38                    script.type = 'text/javascript';
39                    script.charset = 'utf-8';
40                    document.getElementsByTagName("head")
41                    .item(0).appendChild(script);
42                    oldOnloadHandler.apply(this, arguments);

```

```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Recently Rated widget tag

- [Line 1](#) defines the container that holds the Recently Rated widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Recently Rated widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Recently Rated widget inside the container defined in [line 1](#). [Line 8](#) specifies the Recently Rated widget tag's settings, including the type of topics the Recently Rated widget will list. If you want the Recently Rated widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Recently Rated widget is rendered on the page. If the Recently Rated widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Recently Rated widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag to the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

Figure 19: Recently Rated widget displayed on a web page

The screenshot shows a web page for 'force sport'. The main article is 'An Ace is Born' by John Candy, dated January 17, 2009. Below the article is a 'Rate' widget with a star rating and a 'Submit' button. To the right of the article is a 'Recently Rated' widget listing the article 'An Ace is Born'. Annotations point to the 'Recently Rated' widget title and the article title 'An Ace is Born'.

### Note

When the Recently Rated widget is deployed, it does not display a title (as shown in Figure 19). For the Recently Rated widget to display a title, you will have to insert the title into the template to which you copied the Recently Rated widget tag.

## Most Rated Widget Tag

The “Most Rated Deployment” screen (**Ratings > Deploy > Most Rated**) provides administrators and designers with the Most Rated widget tag. This section analyzes the parameters defined in the Most Rated widget tag:

### Most Rated widget tag

```

1     <div id="topics_container_most_rated_other"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "wsdk.topics",
6             version: "1.5",
7             elementID: "_most_rated_other_container",
8             attributes: {"content_type":"most_rated",
9                 "resource_type":"other","site_id":
10                "FirstSiteII","count_topics":"10"}});
11        setTimeout(
12            function(){if ((typeof(wsdk) ==
13                'undefined') || (typeof(wsdk.topics) == 'undefined'))
14                {document.getElementById('topics_container_most_rated
15                _other').innerHTML = "<div style='font-family: Tahoma,
16                Verdana, Geneva, sans-serif;font-size: 12px;color:
17                #333333;border: 1px solid #dbdfef;padding-left:
18                5px;padding-top: 4px;height: 22px;'>Topics is unavailable
19                right now. Please contact the site administrator.</div>";}}
20            ,30000);

21        cos.pageScripts = cos.pageScripts || [];
22        cos.pageScripts.push('wsdk.topics');

23        (function()
24        {
25            var oldOnloadHandler = window.onload || function()
26            {
27                };
28            if (!oldOnloadHandler.alreadyProcessed)
29            {
30                window.onload = function()
31                {
32                    var script = document.createElement
33                    ('script');
34                    script.src = 'http://cosservice.fatwire.com
35                    :8080/cos/wsdk/widget/'
36                    + cos.pageScripts.join(':') +
37                    '.js?site_id=FirstSiteII';
38                    script.type = 'text/javascript';
39                    script.charset = 'utf-8';
40                    document.getElementsByTagName("head")
41                    .item(0).appendChild(script);
42                    oldOnloadHandler.apply(this, arguments);

```

```
30         };
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Most Rated widget tag

- [Line 1](#) defines the container that holds the Most Rated widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Most Rated widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Most Rated widget inside the container defined in [line 1](#). [Line 8](#) specifies the Most Rated widget tag's settings, including the type of topics the Most Rated widget will list. If you want the Most Rated widget to include topics of all types in its list, remove the `"resource_type": "other"` parameter.
- [Lines 9 – 11](#) check if the Most Rated widget is rendered on the page. If the Most Rated widget is not rendered after 30 seconds, then users are informed that there is an error and they should contact the site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Most Rated widget tag (`wsdk.topics`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 20:** Most Rated widget displayed on a web page

The screenshot shows a web page for 'force sport'. The main article is titled 'An Ace is Born with Reviews' by John Gandy, dated January 17, 2009. Below the article is a 'Rate' widget with a 3.0 star rating and a 'Submit' button. The widget lists the article title 'An Ace is Born with Reviews 3.0'. Annotations indicate that the widget title is 'Most Rated' and the article title is 'An Ace is Born with Reviews 3.0'.

**Note**

When the Most Rated widget is deployed, it does not display a title (as shown in [Figure 20](#)). For the Most Rated widget to display a title, you will have to insert the title into the template to which you copied the Most Rated widget tag.

## Login Bar Widget Tag

The “Session Box Deployment” screen (**Login > Deploy**) provides administrators and designers with the Login Bar widget tag. This section analyzes the parameters defined in the Login Bar widget tag:

### Login Bar widget tag

```

1     <div id="session_box_container"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "session-box",
6             version: "0.1",
7             elementID: "session_box_container",
8
9             attributes:{"view_type":"short","site_id":"FirstSiteII"}}});
10        setTimeout(
11            function(){if ((typeof(session_box) ==
12                'undefined') || (typeof(session_box.v0_1) == 'undefined'))
13                {document.getElementById('session_box_container').innerHTML
14                = "<div style='font-family: Tahoma, Verdana, Geneva, sans-
15                serif;font-size: 12px;color: #333333;border: 1px solid
16                #dbdfef;padding-left: 5px;padding-top: 4px;height:
17                22px;'>Login Bar is unavailable right now. Please contact
18                the site administrator.</div>";}}
19            ,30000);

20        cos.pageScripts = cos.pageScripts || [];
21        cos.pageScripts.push('session-box');

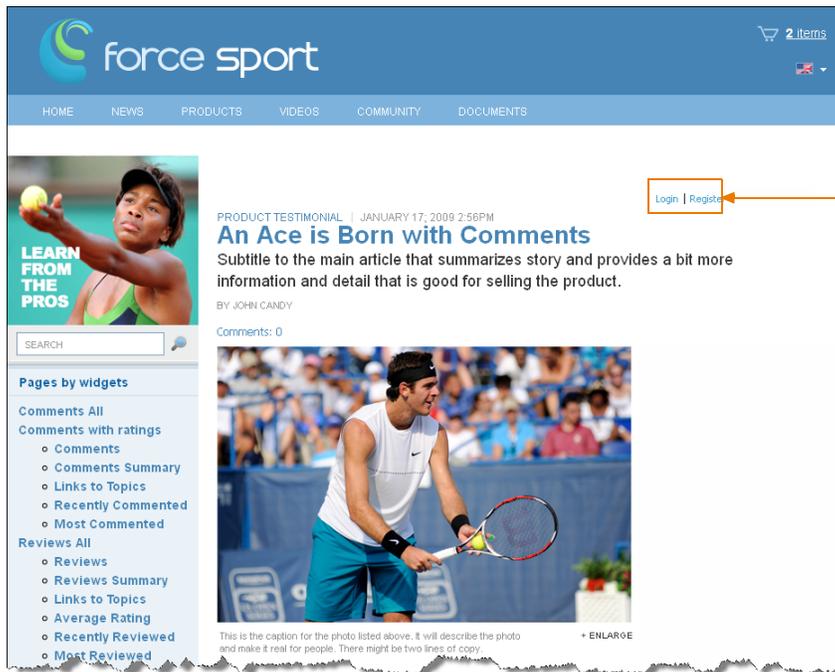
22        (function()
23        {
24            var oldOnloadHandler = window.onload || function()
25            {
26            };
27            if (!oldOnloadHandler.alreadyProcessed)
28            {
29                window.onload = function()
30                {
31                    var script = document.createElement
32                    ('script');
33                    script.src = 'http://cosservice.fatwire.com
34                    :8080/cos/wsdk/widget/'
35                    + cos.pageScripts.join(':') +
36                    '.js?site_id=FirstSiteII';
37                    script.type = 'text/javascript';
38                    script.charset = 'utf-8';
39                    document.getElementsByTagName("head")
40                    .item(0).appendChild(script);
41                    oldOnloadHandler.apply(this, arguments);
42                }
43            }
44        });

```

```
31         window.onload.alreadyProcessed = true;
32     }
33     })();
34 </script>
```

### Analyzing the Login Bar widget tag

- [Line 1](#) defines the container that holds the Login Bar widget on the page.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Login Bar widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Login Bar widget inside the container defined in [line 1](#).
- [Lines 9 – 11](#) check if the Login Bar widget is rendered on the page. If the Login Bar widget is not rendered after 30 seconds, users are informed that there is an error and they should contact their site administrator.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- [Line 13](#) adds the Login Bar widget tag (`session-box`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 21:** Standalone login bar widget displayed on a web page

Standalone **Login Bar** widget. Visitors can use this widget to log in to the web site, register to become an authenticated visitor, and request a new password.

## Poll Widget Tags

The “Deploy *Poll Name* Poll” screen (**Polls > All Polls > navigate to the desired poll > Deploy**) provides administrators and designers with a given poll’s Poll and Results widget tags. This section analyzes the parameters defined in the following widget tags:

- [Main Poll Widget Tag](#)
- [Poll Results Widget Tag](#)

### Main Poll Widget Tag

The “Poll Tag” is the main Poll widget tag which contains all of the configurations for the poll you are deploying.

#### Poll Tag widget tag

```

1      <div id="poll_container1322111446303"></div>
2      <script type="text/javascript">
3          cos = window.cos || {};
4          cos.pageWidgets = cos.pageWidgets || [];

5          cos.pageWidgets.push({name: "poll",
6              version: "1.0",
7              elementID: "poll_container1322111446303",
8              attributes:
9              {"poll_id": "1322111446303", "uid": "35b48e18-f1c7-4e97-ab1a-
10             98cea2564f66", "site_id": "FirstSiteII"}});
11             setTimeout(
12                 function(){if ((typeof(poll) ==
13                 'undefined') || (typeof(poll.v1_0) == 'undefined'))
14                 {document.getElementById('poll_container1322111446303').inn
15                 erHTML = "<div style='font-family: Tahoma, Verdana, Geneva,
16                 sans-serif;font-size: 12px;color: #333333;border: 1px solid
17                 #dbdfef;padding-left: 5px;padding-top: 4px;height:
18                 22px;'>Poll is unavailable right now. Please contact the
19                 site administrator.</div>";}}
20                 ,30000);

21         cos.pageScripts = cos.pageScripts || [];
22         cos.pageScripts.push('poll');

23         (function()
24             {
25                 var oldOnloadHandler = window.onload || function()
26                 {
27                 };
28                 if (!oldOnloadHandler.alreadyProcessed)
29                 {
30                     window.onload = function()
31                     {
32                         var script = document.createElement
33                         ('script');
```

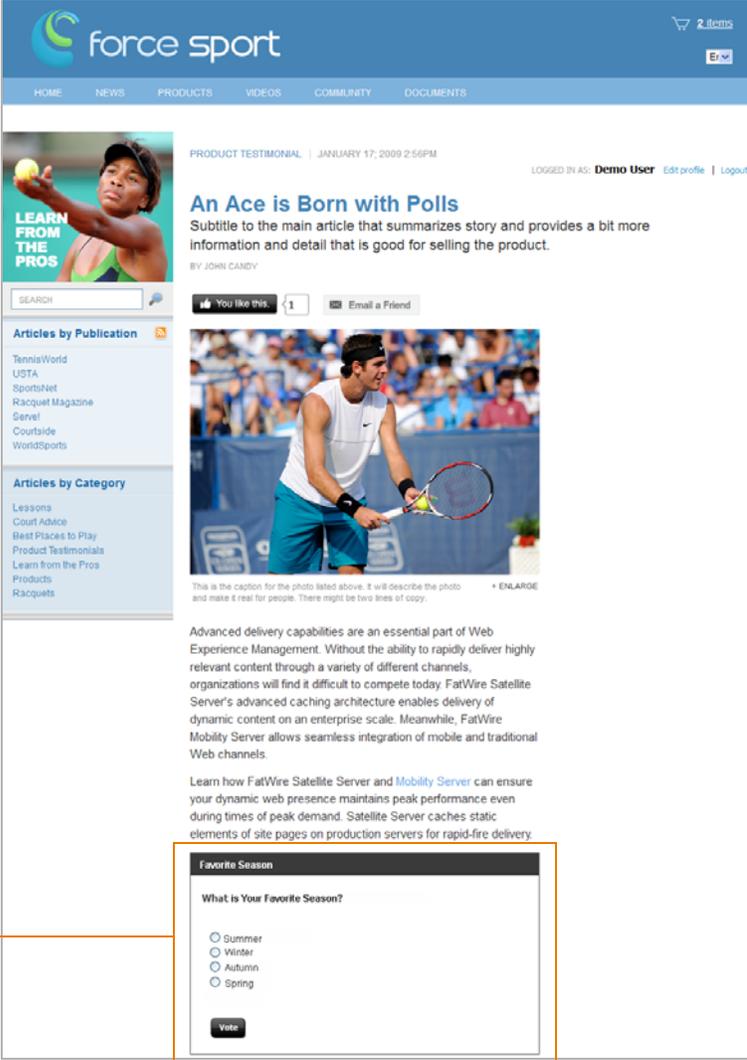
```
24         script.src = 'http://cosservice.fatwire.com
25             :8080/cos/wsdk/widget/'
26             + cos.pageScripts.join(':') +
27             '.js?site_id=FirstSiteII';
28         script.type = 'text/javascript';
29         script.charset = 'utf-8';
30         document.getElementsByTagName("head")
31             .item(0).appendChild(script);
32         oldOnloadHandler.apply(this, arguments);
33     };
34     window.onload.alreadyProcessed = true;
35 }
36
37 }());
38 </script>
```

### Analyzing the Poll widget tag

- **Line 1** defines the container that holds the Poll widget on the page. This line also specifies the unique identifier of the Poll widget you are deploying. Each poll is automatically assigned a unique identifier upon creation.
- **Line 2** opens the bootstrapping JavaScript code needed for the widget.
- **Line 3** defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- **Line 4** defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- **Lines 5 – 8** define the Poll widget tag and add the new element to the list of widget tags (defined in **line 4**). **Line 7** contains a link that is used to render the Poll widget inside the container defined in **line 1**.
- **Lines 9 – 11** check if the Poll widget is rendered on the page. If the Polls widget is not rendered after 30 seconds, users are informed that there is an error and they should contact the site administrator.
- **Line 12** defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's value is empty.
- **Line 13** adds the Poll widget tag (`poll`) to the list of page scripts (defined in **line 12**).
- **Line 16** checks if there is any logic deployed on the page which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- **Line 19** checks if the `onload` function is already processed.
- **Line 21** overrides the code that runs when the page is loaded.
- **Lines 23 – 28** add the JavaScript tag on the page, including a list of all the scripts that need to be deployed.
- **Line 29** adds a new `onload` function call to the old `onload` function, with all necessary arguments.
- **Line 31** defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.

- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 22:** Poll widget displayed on a web page



The screenshot shows a web page for 'force sport'. The main article is titled 'An Ace is Born with Polls' by John Candy, dated January 17, 2009. The article includes a photo of a tennis player and a poll widget titled 'Favorite Season' with the question 'What is Your Favorite Season?'. The poll options are Summer, Winter, Autumn, and Spring, with a 'Vote' button below them. The poll widget is highlighted with an orange box.

**Poll** widget displayed on a web page. This tag displays the poll's question and available answer options. Site visitors can select an answer and click **Vote** to participate in the poll.

## Poll Results Widget Tag

The “Results Tag” contains a summary of the main poll’s results (“[Main Poll Widget Tag](#),” on page 256), which dynamically change with each poll vote submitted by a site visitor.

### Results widget tag

```

1     <div id="poll_summary_container1322111446303"></div>
2     <script type="text/javascript">
3         cos = window.cos || {};
4         cos.pageWidgets = cos.pageWidgets || [];

5         cos.pageWidgets.push({name: "poll-summary",
6             version: "1.0",
7             elementID: "poll_summary_container1322111446303",
8             attributes:
9             {"poll_id":"1322111446303","uid":"35b48e18-f1c7-4e97-ab1a-
10             98cea2564f66","site_id":"FirstSiteII"}}});
11             setTimeout(
12                 function(){if ((typeof(poll) ==
13                 'undefined') || (typeof(poll.v1_0) == 'undefined'))
14                 {document.getElementById('poll_summary_container13221114463
15                 03').innerHTML = "<div style='font-family: Tahoma, Verdana,
16                 Geneva, sans-serif;font-size: 12px;color: #333333;border:
17                 1px solid #dbdfef;padding-left: 5px;padding-top:
18                 4px;height: 22px;'>Poll is unavailable right now. Please
19                 contact the site administrator.</div>";}}
20                 ,30000);

21         cos.pageScripts = cos.pageScripts || [];
22         cos.pageScripts.push('poll');

23         (function()
24         {
25             var oldOnloadHandler = window.onload || function()
26             {
27                 };
28             if (!oldOnloadHandler.alreadyProcessed)
29             {
30                 window.onload = function()
31                 {
32                     var script = document.createElement
33                     ('script');
34                     script.src = 'http://cosservice.fatwire.com
35                     :8080/cos/wsdk/widget/'
36                     + cos.pageScripts.join(':') +
37                     '.js?site_id=FirstSiteII';
38                     script.type = 'text/javascript';
39                     script.charset = 'utf-8';
40                     document.getElementsByTagName
41                     ("head").item(0).appendChild(script);
42                     oldOnloadHandler.apply(this, arguments);
43                 }
44             }
45         });

```

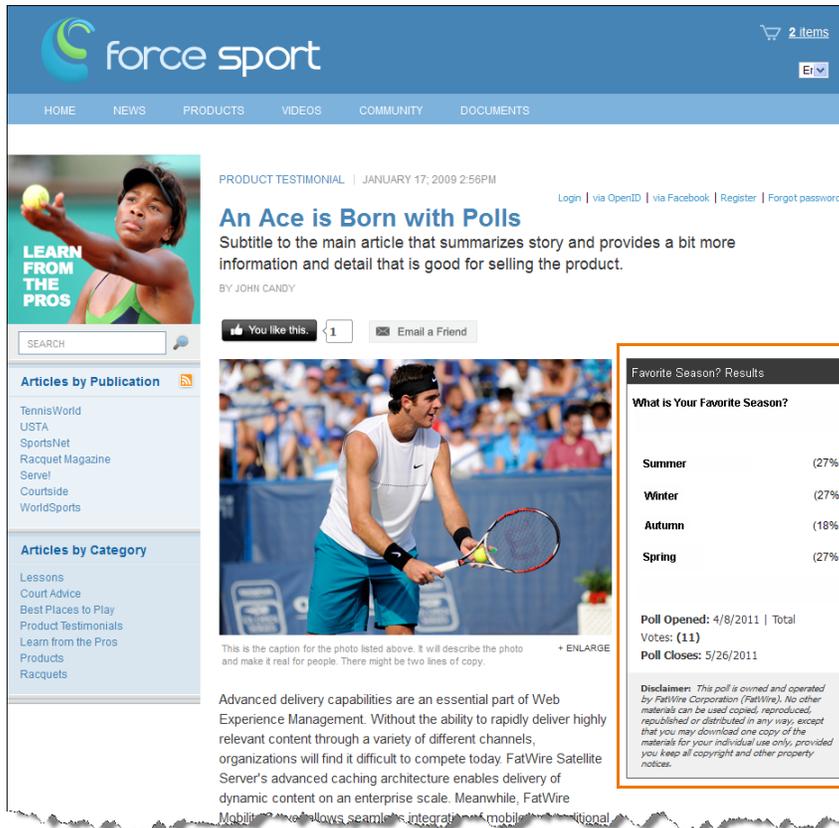
```
31         window.onload.alreadyProcessed = true;
32     }

33     })();
34 </script>
```

### Analyzing the Results widget tag

- [Line 1](#) defines the container that holds the Results widget on the page. This line also specifies the unique identifier of the poll with which the Results widget is associated. The Results widget displays the results of the poll whose unique identifier is specified in this line.
- [Line 2](#) opens the bootstrapping JavaScript code needed for the widget.
- [Line 3](#) defines the `cos` object on the page. The `cos` object contains all Community application functionality. If there are no community widget tags deployed on the page, then the `cos` object's value is empty.
- [Line 4](#) defines the list of widget tags that are deployed on the page. If there are no widgets deployed on the page, then the list's value is empty.
- [Lines 5 – 8](#) define the Results widget tag and add the new element to the list of widget tags (defined in [line 4](#)). [Line 7](#) contains a link that is used to render the Results widget inside the container defined in [line 1](#).
- [Lines 9 – 11](#) check if the poll results widget is rendered on the page. If the poll results widget is not rendered after 30 seconds, users are informed that there is an error and they should contact the site administrators.
- [Line 12](#) defines the list of scripts deployed on the page. If there are no scripts deployed, then the list's values is empty.
- [Line 13](#) adds the Results widget tag (`poll`) to the list of page scripts (defined in [line 12](#)).
- [Line 16](#) checks if there is any logic deployed on the page, which starts after the page has been loaded. If there is logic deployed on the page, this line defines the function that implements it. If there is no such logic deployed, then the function is left empty.
- [Line 19](#) checks if the `onload` function is already processed.
- [Line 21](#) overrides the code that runs when the page is loaded.
- [Lines 23 – 28](#) add the JavaScript tag on the page, including a list of scripts that need to be deployed.
- [Line 29](#) adds a new `onload` function call to the old `onload` function, with all the necessary arguments.
- [Line 31](#) defines the `alreadyProcessed` variable in order to avoid processing the `onload` function again.
- [Line 34](#) closes the bootstrapping JavaScript code needed for the widget.

**Figure 23:** Results widget displayed on a web page



**Results** widget displayed on a web page. This widget displays the results of its associated poll.



## Appendix B

# Enabling SEO Support for Community Widgets

This appendix provides information about SEO support and instructions on how to enable SEO support for your website.

This appendix contains the following sections:

- [Overview](#)
- [Enabling SEO Support](#)

## Overview

Community widget tags are based on JavaScript code. A browser that is set to disable JavaScript does not display any community widgets. Enabling SEO support ensures that visitors' comments and reviews are visible in browsers set to disable JavaScript. However, visitors are limited to having read-only access to comments and reviews. All other widgets and widget functionality remain hidden.

Additionally, SEO support enables search engines such as *Google* to look for search terms within the content of your website's comments and reviews. If a comment or review contains a search term, the search engine will display a link to the web page, on which that comment or review was posted, in its search results list.

# Enabling SEO Support

The fields used to enable SEO support are located at the bottom of the Comments and Reviews widget tag custom deployment screens. Below shows the SEO fields in the Comments widget tag's deployment screen (the Reviews widget tag's custom deployment screen contains the same SEO fields):

**Comments Deployment**

Tag Format:  Site settings,  Custom settings

Comment Type: Default

Resource ID: [text input]

Resource Type: Others

Comment Submit Box: Default

Display Order: Default

Comment Ranking: Default

Resource Title: [text input]

Resource URL: [text input]

Post Status: Open

Login Bar Required: Default

**Tag**

```
<div id="comments_container_other"></div>
<script type="text/javascript">
cos = window.cos || {};
cos.pageWidgets = cos.pageWidgets || {};

cos.pageWidgets.push({name: "wsdk.comments",
version: "1.5",
elementID: "comments_container_other",
attributes:
{"resource_type":"other","comment_post_status":"open","site_id":"f
orMelinda"}});
setTimeout(
function(){if ((typeof(wsdk) ==
'undefined') || (typeof(wsdk.comments) == 'undefined'))
{document.getElementById("comments_container_other").innerHTML
= "<div style='font-family:Tahoma,Verdana,Geneva,
sans-serif;font-size: 12px;color: #000000;border: 1px solid
#d0d0d0;padding-left: 5px;padding-top: 4px;height:
22px;'>Comments is unavailable right now. Please contact the site
administrator.</div>";}
},30000);

cos.pageScripts = cos.pageScripts || [];
cos.pageScripts.push("wsdk.comments");

(function()
{
var oldOnloadHandler = window.onload || function()
{};
if (!oldOnloadHandler.alreadyProcessed)
{
window.onload = function()
{
var script = document.createElement("script");
script.src = "http://cosprod.us.oracle.com/cos/wsdk
/widget/
+ cos.pageScripts.join(";") + ".js?site_id=forMelinda";
script.type = "text/javascript";
script.charset = "utf-8";
document.getElementsByTagName("head").item(0).appendChild(script);
oldOnloadHandler.apply(this, arguments);
};
window.onload.alreadyProcessed = true;
}
})();
</script>
```

**CSS Tag**

```
<link id="cos_css" type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsdk
/skin/wsdk.comments.css?site_id=forMelinda" />
```

**Server-Side Tag for SEO**

```
<#@page import="com.fatwire.cos.widget.tag.CoSWidgetTag">
<%
CoSWidgetTag.builder().widget("wsdk.comments").siteName("forMel
inda").resourceId("${cosId}")cosId("http://cosprod.us.oracle.com
/cos").writer(out).request(request).deploy(); %>
```

**Widget Tag**

Download the SEO widget file

**CSS Tag for SEO**

```
<link type="text/css" rel="stylesheet"
href="http://cosprod.us.oracle.com/cos/wsdk
/skin/wsdk.comments.css?site_id=forMelinda" />
```

Main tag that supports SEO

SEO widget file

CSS tag for SEO support

**To enable SEO support**

1. Access the custom deployment screen for either the Comments or Reviews widget tag. In the menu bar, do one of the following:
  - To access the custom “Comments Deployment” screen, select **Comments > Deploy > Comments > Custom Settings**
  - To access the custom “Reviews Deployment” screen, select **Reviews > Deploy > Reviews > Custom Settings**.
2. In the “Resource ID” field, specify a resource ID for the Comments or Reviews widget. The “Server-Side Tag for SEO” inherits the specified resource ID.
3. Place the `cos-widget-tag.jar` file in the application server’s classpath (for example, `WEB-INF/lib`):
  - a. In the “Widget Tag” field, click **Download the SEO Widget file** to download the SEO widget file (`cos-widget-tag.jar`).
  - b. Place the SEO widget file (`cos-widget-tag.jar`) into the application server’s classpath.

This `jar` file defines the functionality of the “Server-Side Tag for SEO.” Without this `jar` file, SEO support cannot be enabled.

4. Deploy the “Server-Side Tag for SEO” on the page on which the Comments (or Reviews) widget is deployed:
  - a. In the “Server-Side Tag for SEO” field, copy (**Ctrl+C**) the tag.
  - b. Access the template into which you inserted the Comments (or Reviews) widget tag and insert the “Server-Side Tag for SEO” directly below the Comments (or Reviews) widget tag. For instructions, see [step 4 on page 29](#).

The “Server Side Tag for SEO” is the main tag that enables SEO support for your web pages.

5. Deploy the “CSS Tag for SEO.” This tag is required to display the web page’s comments and reviews in the proper layout when a visitor’s browser is set to disable JavaScript. Do the following:
  - a. In the “CSS Tag for SEO” field, copy (**Ctrl+C**) the tag.
  - b. Access the template into which you inserted the Comments (or Reviews) widget tag and insert the “CSS Tag for SEO” between the `<head></head>` parameters of the template’s source code. For instructions, see [step 4 on page 29](#).

The “CSS Tag for SEO” inherits the appearance settings that are configured for the Comments and Reviews widgets. When deployed, this tag applies JavaScript formatting to the comments and reviews listed on the web page.

6. Publish the template. For instructions, see the *Oracle WebCenter Sites Administrator’s Guide*.
7. Verify that you have enabled SEO support properly. Using a browser set to disable JavaScript, access the website and view a web page on which a Comments (or Reviews) widget is deployed.

## Appendix C

# Deploying the CSS Tag

This appendix provides information about the CSS Tag and instructions for deploying this tag on web pages.

This appendix contains the following sections:

- [Overview](#)
- [Deploying the CSS Tag](#)

## Overview

Each widget tag contains its own CSS which applies the desired look and feel to the widget on a web page. A widget's CSS is loaded when a request is made for the web page on which that widget is deployed. If multiple widgets are deployed on the same page, the system will load each widget tag's CSS separately, which increases the load time of the web page.

To decrease the load time of a web page on which multiple widget tags are deployed, the Community application provides the CSS tag. When the CSS tag is deployed, it combines all CSS files of the widgets deployed on that page into one CSS. This means that each widget's CSS will be loaded at the same time, which reduces the amount of requests the system makes to load the page's content. Therefore, decreasing the time it would take the Community application to load all CSS files for the widgets deployed on the page.

## Deploying the CSS Tag

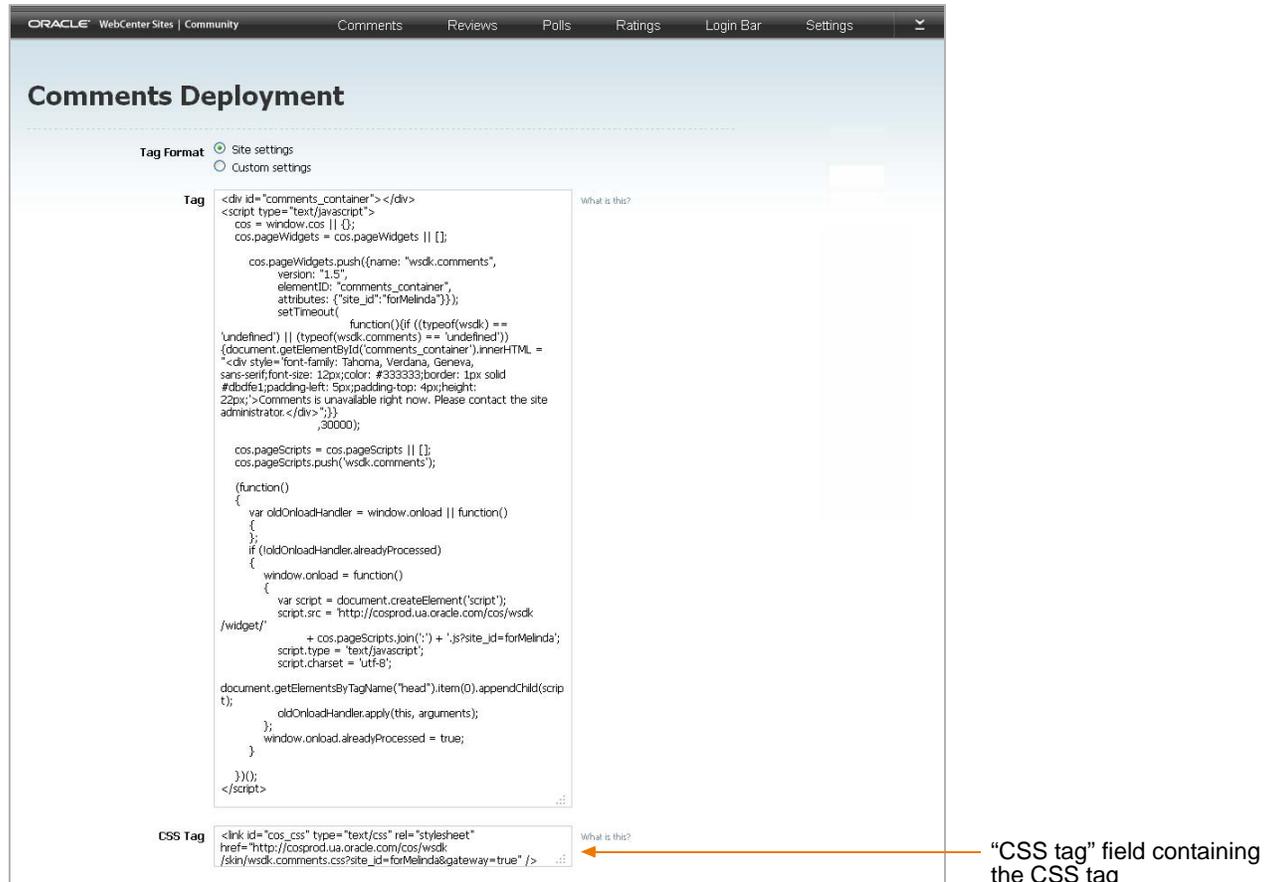
The "CSS Tag" field is available in the deployment screen of any community widget. This scenario uses the CSS tag in the "Comments Deployment" screen (**Comments > Deploy > Comments**).

For the CSS tag to load all CSS files for the widgets deployed on a given web page, you will have to configure the CSS tag by specifying the widget tags and the associated CSS files that will be loaded with the page. When the page is requested, the Community application scans the code of the CSS tag and makes a single request for all widget CSS files that are specified in the CSS tag.

### Note

If a widget that is deployed on the page is not specified inside the CSS tag, the page will not load that widget's CSS.

Below shows the “CSS Tag” field in the “Comments Deployment” screen (all other widget deployment screens contain a similar “CSS Tag” field).



### To configure and deploy the CSS tag

1. Access the deployment screen of any community widget tag. In this example, we access the “Comments Deployment” screen (**Comments > Deploy > Comments**).
2. In the “CSS Tag” field, copy (**Ctrl+C**) the CSS tag.
3. Access the WebCenter Sites Admin interface, and insert the CSS tag into the template that renders the desired web page:
  - a. In the menu bar, point to the down-arrow icon, located at the extreme right of the screen, to render the applications bar.
  - b. In the applications bar, click the WebCenter Sites **Admin** icon to render the WebCenter Sites Admin interface.
  - c. Locate the template into which you wish to insert the CSS tag:
    - 1) From the start menu options, click **Search**.
    - 2) In the “Search” results list, select **Find Template**.
    - 3) In the “Search for Templates” form, click **Search**.

- 4) In the “List of Templates” screen, select the template into which you wish to insert the widget tag.
- 5) In the template’s “Inspect” form, click **Edit**.
- 6) In the template’s “Element” screen, insert (**Ctrl+V**) the widget tag into the “Element Logic” field between the <head></head> tags. For example:

```
<head>
  <link id="cos_css" type="text/css" rel="stylesheet"
    href="http://coservice.fatwire.com:8080/cos/wsdk/
    skin/wsdk.comments.css?site_id=FirstSiteII
    &gateway=true" />
  ...
</head>
```

- d. In the CSS tag, specify each widget tag and CSS file that will be loaded with the page. For example:

If the CSS tag will load the CSS files for the Comments, Reviews, Polls, and Login Bar widgets, the CSS tag should be modified as follows:

```
1 <link id="cos_css" type="text/css" rel="stylesheet"
  href="http://prod:port/cos/wsdk/skin/
  wsdk.comments:wsdk.reviews:poll:session-
  box.css?site_id=FirstSiteII&gateway=true" />
2 where
3 wsdk.comments- name of the comments widget;
4 wsdk.reviews- name of the review's summary widget;
5 poll- name of the poll widget;
6 session-box- name of the login bar widget;
7 wsdk.comments:wsdk.reviews:poll:session-box.css
```

Below provides descriptions for the lines in the CSS tag:

- [Line 1](#) specifies the CSS tag’s href value, which points to the location of the CSS files that will be loaded for the widget tags deployed on the page. Notice that each widget’s name is separated by a colon.
- [Lines 3 – 6](#) individually specify each widget’s name. In this example, [line 3](#) specifies the Comments widget, [line 4](#) specifies the Reviews widget, [line 5](#) specifies the Polls widget, and [line 6](#) specifies the Login Bar widget.
- [Line 7](#) specifies the CSS file of each widget (specified in [lines 3 – 6](#)) that the CSS tag will load on the page.

For detailed information about the CSS tag, see the *Oracle WebCenter Sites Developer’s Guide for the Community Application*.

- e. Click **Save Changes**.
4. Publish the template to the website. For instructions about publishing, see the *Oracle WebCenter Sites Administrator’s Guide*.

# Glossary

This glossary explains the terms used throughout this guide that are specific to the Community application.

## **Anonymous Visitor**

A [Site Visitor](#) who does not have credentials for logging in to the website. These visitors are also referred to as guests.

## **Authenticated Visitor**

A [Site Visitor](#) with credentials for logging in to the website.

## **Auto-Moderation Filters**

The configurable filters that can be applied to assist administrators and moderators with the moderation process (“Restricted Words,” “Visitor Blacklist,” and “Visitor Whitelist”).

## **CM Site**

The content management site where the Community application is enabled. All comments, reviews, ratings, and poll votes submitted by site visitors are collected on this site.

## **Community Widget**

Added site functionality provided by the Community application:

- Comments, Comments Summary, Links to Topics, Recently Commented, and Most Commented widgets
- Reviews, Reviews Summary, Links to Topics, Average Rating, Top Ranked Topics, Recently Reviewed, and Most Reviewed widgets
- Poll and Results widgets
- Stars Ratings, Thumbs Up/Down Ratings, Like It Ratings, Recommend Ratings, Average Rating, Recently Rated, and Most Rated widgets
- Login Bar widget

## Community Widget Tag

The code that defines the widget's properties, visitors' permissions to use the widget, and the system on which visitors' input will be collected and stored. Administrators and designers deploy widget tags on web pages to display the widgets to site visitors.

## CSS Tag

The tag that, when deployed, combines all the CSS files of the widgets deployed on a page into one CSS. This means that each widget's CSS is loaded at the same time, which reduces the amount of requests the system makes to load the page's content. This tag should only be deployed on a page on which multiple widget tags are deployed.

## Custom Settings

The settings of a widget tag that are configured using the tag generator in the widget's deployment screen. Custom settings are temporary values that are saved only in a particular instance of the tag (they cannot be saved in the Community interface). As a result, each time a widget tag's settings are customized, the tag must be redeployed on the desired web pages.

## Default Settings

A widget's saved settings. Default settings for the Comments, Reviews, and ratings (Stars, Thumbs Up/Down, Like It, and Recommend) widget tags can be reconfigured in the "Permissions" and "Appearance" configuration screens specific to the desired widget. When these settings are saved, the default tag is dynamically updated everywhere it is used. The default settings for all other widget tags are configured out-of-the-box and cannot be reconfigured.

## Deployment

The process of inserting community widget tags into the desired template assets.

## Designer

A Community interface user who can configure the default appearance of community widgets, create polls, and customize a given community widget tag with custom values. These users can also specify the domain on which community widget tags can be deployed and use the WebCenter Sites Admin interface to deploy community widget tags and publish the templates.

## General Settings

The moderation, security, and language settings that Community interface users can configure for all community widgets of a given site.

## Manual Moderation

The process of administrators and moderators manually approving, modifying, and/or deleting visitors' comments, reviews, and ratings.

**Moderator**

A Community interface user who can set visitor permissions to community widgets, and configure the moderation settings and the auto-moderation filters for community widgets. These users can also manually moderate comments, reviews, and ratings that are posted to web pages on which community widget tags are deployed.

**SEO (Search Engine Optimization)**

The process of optimizing the JavaScript based Comments and Reviews widgets. This process enables Comments and Reviews widgets to be displayed to site visitors using browsers set to disable JavaScript and allows search engines (such as *Google*) to look for search criteria within the content of the comments and reviews posted to your website.

**Site Administrator**

A site user who has administrative permissions to an entire site and its applications, including the Community application. These users have permissions to all Community interface functions, such as configuring the appearance of community widgets, setting visitor permissions to community widgets, managing general community widget settings, and deploying community widget tags on various web pages.

**Site Visitor**

Any visitor of the website.

